

NOTE: The contents of this PDF contain an export of the 2023-2024 OBIS/OTGA Contributing and publishing datasets to OBIS. Be aware that some guidelines presented in this PDF may be out of date with current recommendations.

See the OBIS Manual (<https://manual.obis.org/>) for up to date guidance.

Additionally, note that the quizzes and assignments that were in the course are not included in this export. We also apologize for any formatting mistakes that may have occurred during export.

Start Here

Site: [OceanTeacher Global Academy](#)
Course: Contributing and publishing datasets to OBIS (self-paced)
Book: Start Here

Table of contents

Getting Started

- Overview
- Course Outline and Agenda
- Meet the Trainers

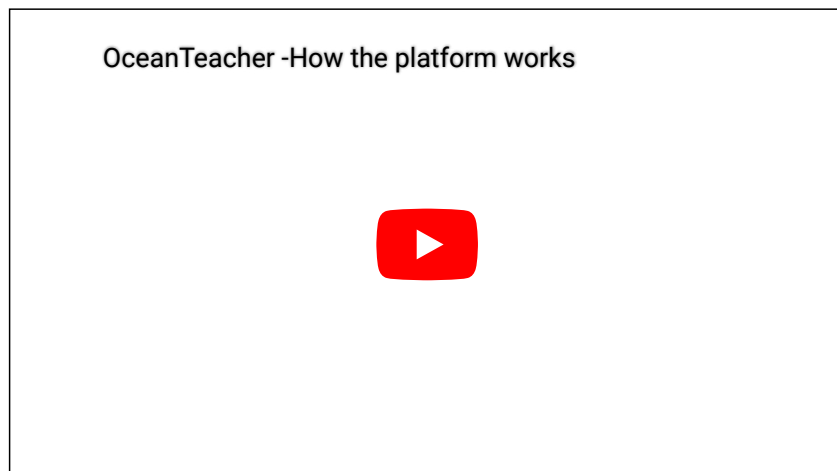
Frequently Asked Questions

- Minimum technology requirements
- How long can I access the course contents/resources online?
- Course Feedback
- Certificate
- Learner Support Resources

Acknowledgements

Welcome to Contributing and publishing datasets to OBIS!

To get started please watch the video below on how to navigate the OceanTeacher e-Learning platform.



When you are done, carefully review the rest of this Getting Started module.

Course Overview

The [Ocean Biodiversity Information System](#) (OBIS) is the most comprehensive gateway to the world's ocean biodiversity and biogeographic data and information. OBIS provides open access to this quality-controlled data that adhere to biodiversity standards (Darwin Core, EML). This course will teach you step by step the basics for how to format and publish datasets according to these standards so that you may contribute to OBIS, and adopt best practices in the management of marine life data.

Learning goals

At the end of this training, learners should be able to:

- Understand Darwin Core (DwC) standards
- Format datasets following DwC standards
- Publish datasets to OBIS using an IPT
- Access data that has been published to OBIS

Course outline

Learning Outcomes

Upon successful completion of this training, learners will be able to:

- Format data tables that adhere to Darwin Core
- Find and use controlled vocabularies
- Publish datasets using Integrated Publishing Toolkits
- Find, access, and download published datasets within OBIS

The course is divided into the following modules:

Module 1: *Introduction to data publishing and Darwin Core*

This module will introduce you to Darwin Core standards and the publishing schema OBIS adheres to.

Module 2: *Introduction to data formatting*

In this module you will learn about the different data structures and when to use them, how to match taxon names to the World Register of Marine Species, how to construct identifiers that link records between tables, and some common formatting challenges, including standardizing dates and coordinates.

Module 3: *Formatting data tables*

In this module you will learn how to format core and extension data tables, and how to map terms to Darwin Core.

Module 4: *DNA derived data*

This module will introduce you to DNA derived data, and how to format it to adhere to Darwin Core.

Module 5: *Controlled Vocabulary*

In this module you will learn about the importance of using controlled vocabulary, and how to select vocabulary terms for measurement identifiers in the ExtendedMeasurementOrFact extension.

Module 6: *Conducting Quality Control*

This module will introduce some quality control checks you can implement to ensure your dataset will have minimal data quality flags.

Module 7: *Publishing your Data*

In this module, you will learn how to publish datasets to OBIS.

Module 8: *Accessing Data from OBIS*

This short module will review the different ways you can access data that has already been published to OBIS.

Assessment

Each module includes exercises and short quizzes to assess your knowledge. To pass the course and receive a certificate of achievement, you must obtain a minimum score of 80% on all quizzes and exercises. Exercises are graded once weekly by the course instructor, so note that you may receive your grade 1-2 weeks after submitting.

Duration

This self-paced course will take approximately 32 hours to complete.

Elizabeth Lawrence, the course instructor, provides a brief introduction to the course in the video below.

Any questions throughout the course may be directed to Elizabeth through either the chat function on the course platform or via email (e.lawrence@unesco.org or helpdesk@obis.org)

Meet the Trainers

Instructors/Trainers

- [Mr. Ward Appeltans](#) (OBIS project manager)
- [Dr. Elizabeth Lawrence](#) (OBIS training officer) (e.lawrence@unesco.org)

This training course has been developed with financial support from NORAD and LifeWatch ERIC.

FAQs

- How do I contact the course instructor?
 - The best way to contact Dr Elizabeth Lawrence is through either the chat function on the course or via email (e.lawrence@unesco.org or helpdesk@obis.org)
- For more OBIS-specific FAQs see the [OBIS Manual FAQ page](#)

Minimum technology requirements

Learning Management System (Moodle)

- Computer with Windows or Mac OS
- Latest version of either Chrome, Edge, Firefox or Safari web browsers
- JavaScript and cookies enabled
- Broadband internet access (minimum bandwidth of 0.5 Mbps (Receive and Send))
- Speakers or headphones, Microphone

Computer skills required

You should be proficient in the following:

- The ability to be self-directed in learning new technology skills (e.g. following a step-by-step tutorial, online video help, or access to support to learn necessary skills)
- Basic computer skills
- Familiarity with manipulating data in Excel
- Familiarity with R an asset
- Finding resources through search engines
- Downloading and installing software

How long can I access the course contents/resources online?

The course contents will remain available online for as long its contents are relevant. Updates may be provided if deemed necessary. You do not need to download any materials. If you would like to download materials, you can do this by clicking the gear on the right of a module, and select Print book (for the entire Module's content) or Print this chapter (to download a single page).

Module 1: Introduction to data publishing and Darwin Core

Done: View



Print book

Print this chapter

Course Feedback

Finding out about the satisfaction of the learner provides feedback to OTGA and the course organisers (RTC/STC and other stakeholders) on how to improve the learning process, administration and logistics arrangements. The results are used exclusively for continuous improvement for future training courses.

All OTGA courses are evaluated using a standardized, anonymous online feedback survey which is used to collect the opinions from the course participants. Each course is assessed against the required learning outcomes, training activities, and arrangements in order to improve the learning processes.

As such, at the completion of each course, each participant is requested to complete the feedback survey on the last day of the course.

The quality of learning services provided by OTGA is underpinned by the certification of the UNESCO/IOC Project Office for IODE as an ISO 29993 Learning Services Provider.

Certificate

This is a self-taught course that includes quizzes designed to help learners assess their own learning at regular intervals. In order to successfully complete the course and award a Certificate at the end, the following is mandatory:

- complete all lessons in each Module of the course;
- complete all quizzes (80% minimum score for each quiz, unlimited attempts for each quiz, 30-minute time limit for each attempt);
- submit the assignments and receive a minimum grade of 80% or 'satisfactory' according to the exercise type.

Note: filling in the course feedback survey is mandatory to obtain the course certificate.

The certificate can be verified by anyone by visiting the link https://classroom.oceanteacher.org/mod/customcert/verify_certificate.php and entering the verification code which appears on the certificate.

Learner Support Resources

Additional support for this course is available at:

- OBIS Secretariat helpdesk@obis.org
- Course instructor e.lawrence@unesco.org
- [OBIS issues GitHub repo](#)
- OBIS [Slack](#) support channel
- [The OBIS manual](#)

Additional support concerning the use of the OceanTeacher e-Learning Platform can be found:

- by checking the video Understand how the OTGA e-Learning Platform works
- by contacting the OTGA Secretariat by email ioc.training@unesco.org (always use the name of the course as email subject)

Acknowledgements

This course was made possible by the input of several people:

- [Abby Benson](#) - OBIS-USA
- [Yi-Ming Gan](#) - ANTARCTIC OBIS
- [Dr. Ana Carolina Peralta](#) - Caribbean OBIS
- OBIS Vocabulary team
- OBIS Secretariat

Module 1: Introduction to data publishing and Darwin Core

Site: [OceanTeacher Global Academy](#)

Course: Contributing and publishing datasets to OBIS (self-paced)

Book: Module 1: Introduction to data publishing and Darwin Core

Table of contents

Module 1

Lesson 1: The who, what, where, why, and how regarding data contributions

- Five steps to publish
- Why publish data & The FAIR Principles
- Data publishing benefits
- How to handle sensitive data

Lesson 2: Introducing biodiversity standards

- Darwin Core
- Darwin Core Classes
- Taxon Terms
- Occurrence Terms
- Record Terms
- Location Terms
- Event & Sampling Terms

Lesson 3: The Darwin-Core Archive: Core tables and Extensions

- Darwin Core-Archive data tables
- Breaking the star schema: ENV-DATA approach
- ExtendedMeasurementOrFact Extension (eMoF)
- Relational databases (optional)
- Reducing Redundancy
- Relevant Resources

End of Module

Introduction

In this module you will learn about the data standards that OBIS adheres to: Darwin Core and Ecological Metadata Language (EML). You will first learn why it is important and how it can be beneficial to publish your data to OBIS, how sensitive data can be handled, and see an introduction to the OBIS data life cycle. The bulk of this module will focus on learning about Darwin Core, as adhering to data standards is part of what makes OBIS an important and community endorsed resource for open access, quality-controlled ocean data.

Learning Outcomes

After successful completion of this module, you should be able to:

- Identify the five steps to publishing data in OBIS, list the types of marine data OBIS accepts
- Understand the four FAIR principles, explain why sharing data is important and why data standards are beneficial
- Identify qualities of sensitive data and know how to generalize sensitive location data
- Understand Darwin Core principles and know where to find Darwin Core terms
- Understand the difference between flat vs relational databases
- Explain how core tables are linked with extension tables
- Explain how star schema works to reduce redundancy
- Understand how Darwin Core extension files are linked to the core file

How to Proceed

To succeed in this Module, you need to complete the following lessons and exercise:

- Lesson 1: The who, what, where, why, and how regarding data contributions
- Lesson 2: Introducing biodiversity standards
- Lesson 3: The Darwin-Core Archive: Core tables and Extensions
- [Exercise 1-1: Map terms to Darwin Core](#)

as well successfully complete Quiz 1 with a score of $\geq 80\%$

- [Quiz 1](#)

What kind of data can you publish to OBIS?

Since 2000, OBIS has accepted, curated and published marine biodiversity data obtained from varied sources and methods. There is a common misconception that OBIS only accepts species occurrence data - however, this is not true! OBIS can accept many types of marine data including:

- Occurrence and distribution
- Presence/Absence
- Abundance, individual count
- Biomass, size, and life stage
- Abiotic measurements
- Biotic measurements
- Sampling methods and study area
- Sample processing protocols
- Genetic data including sequences
- Data originating from historical records and time series
- Tracking data
- Habitat data
- Acoustic data
- Imaging data
- Metadata describing the dataset and any affiliated project or programme and personnel
- And many other variables associated to the sighting of marine organisms

If you have any of these types of marine data you can contribute these to OBIS! OBIS repository and standards are prepared to cover all data linked to marine life and its community of practice. OBIS accepts data from any organization, consortium, project or individual who wants to contribute data. OBIS Data Sources are the authors, editors, and/or organizations that have published one or more datasets through OBIS, they remain the owners or custodians of the data, not OBIS!

Five steps to publish

OBIS harvests and publishes data from recognized Integrated Publishing Toolkits (IPTs, "digital repositories") from OBIS nodes or [GBIF](#) publishers. If you own data or have the right to publish data in OBIS, which means the data will become openly available, you can contact the [OBIS secretariat or one of the OBIS nodes](#) (or additionally a GBIF publisher). Your organization or programme can also [become an OBIS node](#). An OBIS node usually publishes data from multiple data holders, effectively being a connection in a network of data providers. You may have to first find a [relevant node](#) before you get your data ready to publish. OBIS nodes can cover countries, regions, themes, and/or communities. If you cannot decide on who to contact, please, reach out to the OBIS Secretariat for support.

To publish a dataset to OBIS, there are **five** main steps you must go through:

1. **Identify** which OBIS node is best suited to host your published data. If you would like to publish to GBIF at the same time, that is also possible. Details about this will be covered in Module 7.
2. **Determine the structure of your data** and which format will best suit your dataset. OBIS follows Darwin Core Archive (DwC-A) standards for datasets, and currently adopts a star schema format for data organization. This format is based on relational databases. If you are unfamiliar with such database structures, or would like to refamiliarize yourself with them, we will review that later in this module.
3. **Format your data** according to OBIS and DwC-A standards and guidelines. We will cover formatting in Modules 2, 3, and 4, with specific guides on using controlled vocabularies in Module 5.
4. **Run a series of quality control measures** to ensure you are not missing any required information and that all standards are being met. This helps ensure all data published in OBIS is formatted in a standardized way. When published in OBIS, OBIS provides a quality report to inform data owners and users of any quality control issues. By completing quality control before you publish your dataset you ensure there are fewer errors to fix later. We will review quality control steps in Module 6.
5. Once your dataset is ready for publishing, the **relevant metadata** must be filled in, and then **published on the previously identified IPT**. Both topics will be covered in Module 7.

Why publish data & the FAIR Principles

It is important to publish and ensure your dataset follows a universal standard for several reasons. The [FAIR guiding principles](#) for scientific data management and stewardship provide a good framework to understand the reasoning behind publishing standardized data. FAIR stands for Findable, Accessible, Interoperable, and Reusable. Let's understand each aspect within the FAIR framework and how it is linked to publishing data in OBIS.

- **F - Findable**

Even if you publish your dataset on its own, publishing your data with OBIS will make your data more Findable (and Accessible) to a wider audience you might not have otherwise reached. By publishing your dataset to OBIS you are adding to a global database where your data can be found and analyzed alongside thousands of other datasets. For example, a dataset on [marine invasive species in Venezuela](#) was published July 20, 2022 and as of October 5, 2022 records of this dataset were included in 1,873 data download requests. This can save you time rather than handling individual data requests. OBIS can also provide a Digital Object Identifier (DOI) to your dataset, which means that it can be easily found and tracked throughout the different information systems.

- **A - Accessible**

Similar to being Findable, OBIS makes your datasets more Accessible. Each dataset is given an identifier when you upload it on an IPT (covered in Module 7). Thus when users obtain data from OBIS, the original dataset can easily be identified, accessed, and downloaded. Data from OBIS is accessible in numerous ways (covered in Module 8), giving data users multiple avenues to potentially access your data.

- **I - Interoperable**

Using a standardized data format with controlled vocabularies will ensure your data are more Interoperable - more easily interpreted and processed by computers and humans alike. Increasingly, scientists use computer programs to conduct e-Science and collect data with algorithms. Formatting your data for OBIS will ensure it can be read and accessed by such programs as well as understood by users. OBIS standards also facilitates the connection between your data and other ocean and geospatial information systems.

- **R - Reusable**

Publishing your data allows it to be Reused according to your chosen data usage license. Very likely you expended a lot of resources and time to collect your data and it would be a waste to leave your unique data inaccessible for current and future generations. Likewise, it is better to preserve any data processing done to ensure your dataset is reproducible and/or verifiable. Finally, data in OBIS is often used in several assessment processes and used as information to support policymakers around the globe in making informed decisions.

Data publishing benefits

There are many other benefits of submitting data to OBIS, even if you haven't published any other work on it yet (e.g. scientific papers). This includes:

- Your dataset can be associated with a DOI, allowing for your dataset to be more easily tracked and cited. By ensuring your dataset citation is complete you will ensure you are being acknowledged properly. DOI generation will be covered in Module 7.
- Publishing your dataset with OBIS makes it easier to set it up as a [Data paper](#) or to add your data as supplementary material to your manuscripts, which generates value for you and other researchers.
- When following OBIS publishing requirements, your data will adhere to FAIR standards. OBIS repository also ensures digital safety and promotes a legacy for future generations, not always possible in local repositories.
- You have the means and tools to control modifications in your data and decide when it is published publicly, making it possible to incorporate project or institutional embargos.
- There are social benefits to data publishing as your work becomes integrated into a wider dataset and a community of data providers. It gives both you and your data more visibility. This can lead to more opportunities for collaboration and further career development as a researcher or professional.
- Your data can be incorporated into larger analyses to better understand global ocean biodiversity, helping to shape regional and international policies.
- OBIS provides a large set of tools to further visualize and process your data once published. These scripts and guides markedly optimize our capacity to deliver meaningful information to society.

What about sensitive data?

Sometimes your dataset may contain sensitive information (e.g., location data on endangered or poached species), or perhaps your organization does not want certain details publicly accessible. Types of sensitive data include:

- Location data on endangered or protected species
- Information regarding a commonly poached species
- Species or locations that have an economic impact (positive or negative)

To accommodate sensitivity but still be able to contribute to OBIS, we suggest:

- Generalizing location information by: Obtaining regional coordinates using [MarineRegions](#), [Getty Thesaurus of Geographic Names](#), or [Google Maps](#)
- Using the [OBIS Map tool](#) to generate a polygon area with a Well-Known Text (WKT) representation of the geometry to paste into the `footprintWKT` field
- Delay timing of publication
- Submit your dataset, but mark it as private in the IPT so it is not published right away (i.e., until you set it as public). Alternatively, you can set a password on your dataset in order to share with specific individuals. Note that setting passwords will require some coordination with the IPT manager. By submitting your data to an IPT but not immediately publishing it, you can ensure that the dataset will be in a place to be incorporated at a later date when it is ready to be made public. This not only saves time and helps retain details while relatively fresh in your mind, but also ensures the dataset is still ready to be mobilized in case jobs are changed at a later date.

Details, procedures, and use for each of these tools will be covered in Modules 6 and 7. GBIF has also created the following [Best Practices for Generalizing Sensitive data](#) which can provide you with additional guidance, citation below:

Chapman AD (2020) Current Best Practices for Generalizing Sensitive Species Occurrence Data. Copenhagen: GBIF Secretariat. <https://doi.org/10.15468/doc-5jp4-5g10>.

Introduction to biodiversity standards

From the very beginning, OBIS has championed the use of international standards for biogeographic data. Without agreement on the application of standards and protocols, OBIS would not have been able to build a large central database. OBIS uses the following standards:

- Darwin Core
- Ecological Metadata Language
- Darwin Core Archive and dataset structure

For those wishing to publish biodiversity-based data to OBIS (or GBIF which uses similar standards), following the above data standards will make published data more easily discoverable, fit for use, less ambiguous and more understandable by data users, more interoperable by APIs and data analysis tools, more easily integrated into online resources, and ensure the data adheres with [FAIR data principles](#).

The following pages of this lesson will review each of the above data standards in turn. We will learn how to actually apply these standards to format data in Modules 2 and 3.

Introduction to Darwin Core

[Darwin Core](#) (DwC) is a body of standards (i.e., identifiers, labels, definitions) that facilitate sharing biodiversity informatics and the adaptation of data about life to the digital world. It provides stable [terms](#) and vocabularies related to biological objects/data and their collection, being built and continuously updated by a large international community of data providers and practitioners. Darwin Core is maintained by [TDWG \(Biodiversity Information Standards, formerly The International Working Group on Taxonomic Databases\)](#). Stable terms and vocabularies are important for ensuring the datasets in OBIS remain adequate for use by ensuring consistency of interpretation. By following Darwin Core standards, both data providers and users can be certain of the definition and quality of data.

A Brief History on DwC and OBIS

The old [OBIS schema](#) was an OBIS extension to Darwin Core 1.2., which was based on [Simple Darwin Core](#), a subset of Darwin Core that does not allow any structure beyond rows and columns. This old schema added some terms which were important for OBIS, but were not supported by Darwin Core at the time (e.g., start and end date and start and end latitude and longitude, depth range, lifestage, and terms for abundance, biomass and sample size).

In 2009, the Executive Committee of TDWG announced their ratification of an updated version of Darwin Core as a [TDWG Standard](#). Ratified Darwin Core unifies specializations and innovations emerging from diverse communities, and provides guidelines for ongoing enhancement. The [Darwin Core Quick Reference Guide](#) links to TDWG's term definitions and related practices for Ratified Darwin Core.

In December 2013, the [3rd session of the IODE Steering Group for OBIS](#) agreed to transition OBIS globally to the TDWG-Ratified version of Darwin Core, and the mapping of the (old) OBIS specific terms to Darwin Core can be found [here](#).

Next we will discuss the relevance of various DwC term classes.

Darwin Core Classes

DwC terms correspond to the **column names** of your dataset (as in an Excel table or sample spreadsheet) and can be grouped according to class type for convenience, e.g., Taxa, Occurrence, Record, Location, etc. It is important to use DwC field names because only columns using the exact Darwin Core terms as headers will be recognized by OBIS. We'll see more about these mappings during publishing in Module 7.

There are currently seven required and one strongly recommended DwC terms for publishing datasets to OBIS:

1. `occurrenceID`
2. `eventDate`
3. `decimalLongitude`
4. `decimalLatitude`
5. `scientificName`
6. `scientificNameID` (strongly recommended)
7. `occurrenceStatus`
8. `basisOfRecord`.

A list of all possible Darwin Core terms and their definitions can be found in the [Darwin Core Quick Reference Guide on TDWG](#). We recommend bookmarking this page as it can be a valuable resource to refer back to when formatting different datasets. You can add columns to your dataset according to the available DwC terms, and map your datasheet column names to DwC terms. Note that mapping columns may be one to one, one to many, or many to one, depending on how data was originally recorded. We'll see examples of this throughout the course.

However, OBIS does not currently parse *all* DwC terms. This doesn't mean you cannot include them in your dataset, just know that they may not be parsed when you publish to OBIS - they *will* remain in your source dataset though. There is a convenient [checklist of OBIS-accepted terms](#) available in the OBIS manual.

In the next few pages, we will review most of the relevant Darwin Core terms within different classes to consider when contributing to OBIS, along with guidelines regarding their use. Understanding the meaning of each term will help you organize and format your dataset in a way that not only you, but many other users will be able to interpret it.

There are seven main Darwin Core term class types that are important for OBIS: Taxon, Identification, Occurrence, Record, Location, Event, and Material Sample. We will review each of these classes, the important terms within them, and the associated guidelines for their use.

Taxonomy and Identification

We begin by reviewing DwC terms related to the classes *Taxon* and *Identification* because these class types usually contain associated information. Note there is no relationship between terms presented below, they are presented side by side for ease of reference.

Taxon	Identification
<ul style="list-style-type: none">scientificName	<ul style="list-style-type: none">identifiedBy
<ul style="list-style-type: none">scientificNameID	<ul style="list-style-type: none">dateIdentified
<ul style="list-style-type: none">scientificNameAuthorship	<ul style="list-style-type: none">identificationReferences
<ul style="list-style-type: none">kingdom	<ul style="list-style-type: none">identificationRemarks
<ul style="list-style-type: none">taxonRank	<ul style="list-style-type: none">identificationQualifier
<ul style="list-style-type: none">taxonRemarks	<ul style="list-style-type: none">typeStatus

scientificName (required term) should always contain the originally recorded full scientific name, even if the name is currently a synonym. This is necessary to be able to track back records to the original dataset. The name should include authorship and date information if known and should be the lowest possible taxonomic rank that can be determined, preferably at species level or lower, but higher ranks, such as genus, family, order, class etc. are also acceptable. You may still populate **scientificNameAuthorship** for authorship, but it is preferred to include authorship in **scientificName** as well.

The **scientificName** term should only contain the name and not identification qualifications (such as ?, confer or affinity), which should instead be supplied in the **identificationQualifier** term, see examples below. **taxonRemarks** can capture comments or notes about the taxon or name.

A [WoRMS](#) LSID should be added in **scientificNameID** (a strongly recommended term for each occurrence record). We review how to navigate the WoRMS interface and obtain LSIDs in Module 2. OBIS will use the WoRMS identifier to pull the taxonomic information from the World Register of Marine Species (WoRMS) into OBIS and attach it to your dataset. This information includes:

- Taxonomic classification (kingdom through species)
- The accepted name in case of invalid names or synonyms
- AphiaID
- IUCN red list category

LifeScience Identifiers (LSIDs) are persistent, location-independent, resource identifiers for uniquely naming biologically significant resources. More information on LSIDs can be found at www.lsid.info. For example, the WoRMS LSID for *Solea* *solea* is: urn:lsid:marinespecies.org:taxname:127160, and can be found at the top of each WoRMS taxon page, e.g. [Solea solea](#).

kingdom and **taxonRank** can help us in identifying the provided **scientificName** in case the name is not available in WoRMS. **kingdom** in particular can help us find alternative genus-species combinations and avoids linking the name to homonyms (and is recommended by both OBIS and GBIF). Please contact the WoRMS data management team (info@marinespecies.org) in case the **scientificName** is missing in WoRMS. **kingdom** and **taxonRank** are not necessary when a correct **scientificNameID** is provided.

OBIS recommends providing information about how identification was made, for example by which ID key, species guide or expert; and by which method (e.g morphology vs. genomics), etc. The person's name who made the taxonomic identification can go in **identifiedBy** and *when* in **dateIdentified**. Use the ISO 8601:2004(E) standard for date and time (we will review use of this standard later). A list of references, such as field guides used for the identification can be listed in **identificationReferences**. Any other information, such as identification methods, can be added to **identificationRemarks**.

If the record represents a nomenclatural type specimen, the term **typeStatus** can be used, e.g. for holotype, syntype, etc.

In case of uncertain identifications, and the scientific name contains qualifiers such as *cf.*, *?* or *aff.*, then this name should go in **identificationQualifier**, and **scientificName** should contain the name of the lowest possible taxon rank that refers to the most accurate identification. E.g. if the specimen was accurately identified down to genus level, but not species level, then the **scientificName** should contain the name of the genus, the **scientificNameID** should contain the LSID of the genus, and the **identificationQualifier** should contain the uncertain species name combined with *?* or other qualifiers. The table below shows a few examples:

The use and definitions for additional NO signs (*identificationQualifier*) can be found in [Open Nomenclature in the biodiversity era](#), which provides examples for using the main Open Nomenclature qualifiers associated with *physical specimens*. The publication [Recommendations for the Standardisation of Open Taxonomic Nomenclature for Image-Based Identifications](#) provides examples and definitions for *identificationQualifierS* for *non-physical specimens (image-based)*.

Examples

The following example is from [Benthic fauna around Franz Josef Land](#).

scientificNameID	scientificName	kingdom	phylum	class	order	family	genus	specificEpi
urn:lsid:marinespecies.org:taxname:142004	Yoldiella nana	Animalia	Mollusca	Annelida	Nuculanoida	Yoldiidae	Yoldiella	nana
urn:lsid:marinespecies.org:taxname:140584	Ennucula tenuis	Animalia	Mollusca	Annelida	Nuculanoida	Nuculidae	Ennucula	tenuis
urn:lsid:marinespecies.org:taxname:131573	Terebellides stroemii	Animalia	Annelida	Polychaeta	Terebellida	Trichobranchidae	Terebellides	stroemii

The examples below demonstrate how to capture taxonomic uncertainty with DwC terms:

scientificName	scientificNameID	taxonRank	identificationQualifier	verbatimIdentification
Pelagia Péron & Lesueur, 1810	urn:lsid:marinespecies.org:taxname:135262	genus	gen. nov.	Pelagia gen. nov.
Pelagia benovici Piraino, Aglieri, Scorrano & Boero, 2014	urn:lsid:marinespecies.org:taxname:851656	species	sp. nov.	Pelagia benovici sp. nov.
Gadus Linnaeus, 1758	urn:lsid:marinespecies.org:taxname:125732	genus	cf. morhua	Gadus cf. morhua
Aristeidae Wood-Mason in Wood-Mason & Alcock, 1891	urn:lsid:marinespecies.org:taxname:106725	family	stet.	Aristeidae stet.

Occurrence terms

The following Dwc terms are related to the Class *Occurrence*:

- occurrenceID
- occurrenceStatus
- recordedBy
- individualCount (OBIS recommends to add measurements to ExtendedMeasurementOrFact, eMoF)
- organismQuantity (OBIS recommends to add measurements to eMoF)
- organismQuantityType (OBIS recommends to add measurements to eMoF)
- sex (OBIS recommends to add measurements to eMoF)
- lifeStage (OBIS recommends to add measurements to eMoF)
- behavior
- associatedTaxa
- occurrenceRemarks
- associatedMedia
- associatedReferences
- associatedSequences
- catalogNumber
- preparations

occurrenceID (required term) is an identifier for the occurrence record and should be persistent and globally unique. If the dataset does not yet contain (globally unique) occurrenceIDs, then they should be created. We will review identifier creation in Module 2.

occurrenceStatus (required term) is a statement about the occurrence or sighting of a taxon at a location and at a specific time. It is an important term because it allows us to distinguish between presence and absence records. It is a required term and should be filled in with either **present** or **absent**.

A few terms related to quantity: **organismQuantity** and **organismQuantityType**, have been added to the TDWG ratified Darwin Core. This is a lot more versatile than the older **individualCount** field. However, OBIS recommends the use of the extendedMeasurementOrFact extension for quantitative measurements because of the standardization of terms and the fact that you can link them to sampling events and factual sampling information. We will review use of this extension in Module 3.

In the case where specimens were collected and stored (e.g. museum collections), the **catalogNumber** and **preparations** terms provide the identifier for the record in the collection and to document the preparation and preservation methods. The term **typeStatus** can be used in this context too (listed on the previous page as a Dwc Class:Identification term).

associatedMedia, **associatedReferences** and **associatedSequences** can contain global unique identifiers or URIs pointing to respectively associated media (e.g. online image or video), associated literature (e.g. DOIs), or genetic sequence information (e.g. GenBANK ID).

associatedTaxa include a list (concatenated and separated) of identifiers or names of taxa and their associations with the Occurrence, e.g. the species occurrence was associated with the presence of kelp such as *Laminaria digitata*.

Data columns recording an organism's sex, life stage, and/or behaviour should be populated with controlled vocabulary. It is recommended to include the columns in preferably both the Occurrence table and the extendedMeasurementOrFact table. The recommended vocabulary for **sex** can be found in [BODC vocabulary collection S10](#), for **lifeStage** in [BODC vocabulary collection S11](#), and for **behavior** in [ICES Behaviour collection](#). We will go over use of controlled vocabularies in more detail in Module 5.

occurrenceRemarks can hold additional information about the Occurrence.

recordedBy can hold a list (concatenated and separated) of names of people, groups, or organizations responsible for recording the original Occurrence. The primary collector or observer, especially one with a personal identifier (recordedByID), should be listed first.

Record level terms

The following Dwc terms are related to the Class *Record level*:

- `basisOfRecord`
- `institutionCode`
- `collectionCode`
- `collectionID`
- `bibliographicCitation`
- `modified`
- `dataGeneralizations`

`basisOfRecord` (required term) specifies the nature of the record, i.e. whether the occurrence record is based on a stored specimen or an observation. A full list of vocabularies with definitions for this term can be found [here](#). In case the sampled specimen is stored in a collection (e.g. at a museum, university, research institute), the options are:

- `PreservedSpecimen` (e.g. preserved in ethanol, tissue etc.)
- `FossilSpecimen` (fossil, which allows OBIS to make the distinction between the date of collection and the time period the specimen was assumed alive)
- `LivingSpecimen` (an intentionally kept/cultivated living specimen e.g. in an aquarium or culture collection)

In case no specimen is deposited, the `basisOfRecord` can be `HumanObservation` (e.g. bird sighting, benthic sample but specimens were discarded after counting), `MachineObservation` (e.g. for occurrences based on automated sensors such as image recognition, etc.), or `MaterialSample` (e.g. physical sample was taken, and may have been preserved or destroyed). For records pertaining to genetic samples, `basisOfRecord` should be `MaterialSample` (e.g. in the DNA-derived data extension).

When the `basisOfRecord` is either a `preservedSpecimen`, `LivingSpecimen`, or `FossilSpecimen` please also add the `institutionCode`, `collectionCode`, and `catalogNumber`, which will enable people to visit the collection and re-examine the material. Sometimes, in the case of living specimens, a dataset can contain records pointing to the origin, the in-situ sampling position as well as a record referring to the ex-situ collection. In this case please add the event type information in `eventRemarks`.

`institutionCode` identifies the custodian institute (often by acronym), `collectionCode` identifies the collection or dataset within that institute. Collections cannot belong to multiple institutes, so all records within a collection should have the same `institutionCode`. The `collectionID` is an identifier for the record within the dataset or collection.

`bibliographicCitation` allows for providing different citations on record level, while a single citation for the entire dataset can and should be provided in the metadata. The citation at record level can have the format of a chapter in a book, where the book is the dataset citation. The record citation will have preference over the dataset citation. We do not, however, recommend that you create different citations for every record, as this will explode the number of citations and will hamper the re-use of data.

`modified` is the most recent date-time on which the resource was changed. It is required to use the ISO 8601:2004(E) standard, which will be discussed in Module 2 and Module 6.

`dataGeneralizations` refers to actions taken to make the shared data less specific or complete than in its original form. Populating this field suggests that alternative data of higher quality may (or may not) be available on request. This can be the case for occurrences of vulnerable or endangered species where their positions are converted to the center of grid cells.

Location terms

The following DwC terms are related to the Class *Location*:

- decimalLatitude
- decimalLongitude
- coordinateUncertaintyInMeters
- geodeticDatum
- footprintWKT
- minimumDepthInMeters
- maximumDepthInMeters
- locality
- waterBody
- islandGroup
- island
- country
- locationAccordingTo
- locationRemarks
- locationID

`decimalLatitude` and `decimalLongitude` (required terms) are the geographic latitude and longitude (in decimal degrees), using the spatial reference system given in `geodeticDatum` of the geographic center of a *Location*. The number of decimals should be appropriate for the level of uncertainty in `coordinateUncertaintyInMeters` (at least within an order of magnitude). For `decimalLatitude`, positive values are north of the Equator, negative values are south of it. All values lie between -90 and 90, inclusive. Regarding `decimalLongitude`, positive values are east of the Greenwich Meridian, and negative values are west of it. All values lie between -180 and 180, inclusive.

`coordinateUncertaintyInMeters` is the radius of the smallest circle around the given position containing the whole location

In OBIS, the spatial reference system to be documented in `geodeticDatum` is [EPSG:4326](#). Coordinates in degrees/minutes/seconds can be converted to decimal degrees using the OBIS [coordinates tool](#). We also provide a [tool](#) to check coordinates or to determine coordinates for a location (point, transect, or polygon) on a map. This tool also allows geocoding location names using [marineregions.org](#).

The name of the place or location can be provided in `locality`, and if possible linked by a `locationID` using a persistent ID from a gazetter, such as the MRGID from [MarineRegions](#). If the species occurrence only contains the name of the `locality`, but not the exact coordinates, we recommend using a geocoding service to obtain the coordinates. [Marine Regions](#) has a [search interface](#) for geographic names, and provides coordinates and often precision in meters, which can go into `coordinateUncertaintyInMeters`. Another option is to use the [Getty Thesaurus of Geographic Names](#) or [Google Maps](#). We will go over details on how to deal with uncertain locations in Module 6.

Additional information about the locality can also be stored in DwC terms such as `waterBody`, `islandGroup`, `island` and `country`. `locationAccordingTo` should provide the name of the gazetteer that is used to obtain the coordinates for the locality.

`locationID` is an identifier for the set of location information (e.g. station ID, or MRGID from [marineregions](#)), for example, the [Balearic Plain](#) has MRGID: <http://marineregions.org/mrgid/3956>.

A [Well-Known Text](#) (WKT) representation of the shape of the location can be provided in `footprintWKT`. This is particularly useful for tracks, transects, tows, trawls, habitat extent or when an exact location is not known. WKT strings can be created using our [WKT tool](#). This tool also calculates a midpoint and a radius, which can then be added to `decimalLongitude`, `decimalLatitude`, and `coordinateUncertaintyInMeters` respectively. There is also an [R tool](#) to calculate the centroid and radius for WKT polygons. [wktmap.com](#) can be used to visualize and share WKT strings.

Layers

Switch layers on or off.

EEZ IHO

WKT

Generate WKT.

WKT

Coordinates

Add a location using decimal longitude and latitude.

Enter coordinates

Geocoding

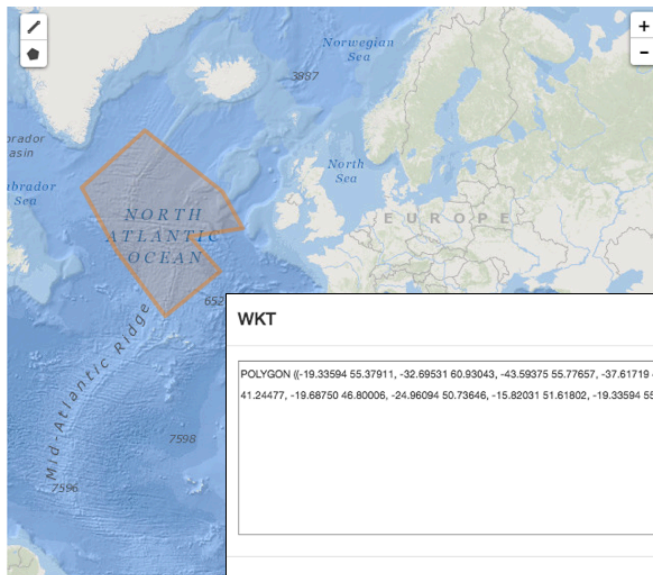
Find locations by name and add them to the locations list.

Enter location name

Type	Name	Longitude	Latitude
No results			

Locations

	Longitude	Latitude	Radius	Name	EEZ	IHO
<input type="checkbox"/>	-27.6095	51.7959	1,179,295			<input checked="" type="checkbox"/>



Keep in mind while filling in `minimumDepthInMeters` and `maximumDepthInMeters` that this should be the **depth at which the sample was taken** and not the water column depth at that location. When filling in any depth fields (`minimumDepthInMeters`, `maximumDepthInMeters`, `minimumDistanceAboveSurfaceInMeters`, and `maximumDistanceAboveSurfaceInMeters`), you should also consider which information is needed to fully understand the data. For a discussion and diagram of different potential scenarios and how to record depth values, please see the [Location section of the OBIS Manual](#).

Event and Sampling Terms

The only term associated with class *MaterialSample* is *materialSampleID*. However, the following DwC terms are related to the Class *Event*:

- parentEventID
- eventID
- eventDate
- type
- habitat
- samplingProtocol (OBIS recommends to add sampling facts to eMoF)
- sampleSizeValue (OBIS recommends to add sampling facts to eMoF)
- SampleSizeUnit (OBIS recommends to add sampling facts to eMoF)
- samplingEffort (OBIS recommends to add sampling facts to eMoF)

eventID is an identifier for the sampling or observation event. **parentEventID** is an identifier for a parent event, which is composed of one or more sub-sampling (child) events (eventIDs). We will review details on how these terms can be constructed in Module 2. But know that some examples of eventIDs include: STAR_arcticsea_st3520_1989-04-04_s01, cruise1_station1_tow1.

The date and time at which an occurrence was recorded goes in **eventDate**. For now all you need to know is this term uses the [ISO 8601 standard](#) and OBIS recommends using the extended ISO 8601 format with hyphens (e.g. 2021-03-21). We will cover more specific guidelines on date formatting in Module 2, as well as in Module 6.

habitat is a category or description of the habitat or ecosystem in which the Event occurred and the organism was sighted (e.g. benthos, seamount, hydrothermal vent, seagrass, rocky shore, intertidal, ship wreck etc.). This information can also be recorded in the extendedMeasurementOrFact extension table ([eMoF](#)) using controlled vocabulary.

Information on **sampleSizeValue** and **sampleSizeUnit** is very important when an organism quantity is specified. However, with OBIS-ENV-DATA it was felt that the eMoF extension would be better suited than the DwC Event Core to store the sampled area and/or volume because in some cases sampleSize by itself may not be detailed enough to allow interpretation of the sample. For instance, in the case of a plankton tow, the volume of water that passed through the net is relevant. In case of Niskin bottles, the volume of sieved water is more relevant than the actual volume in the bottle. In these examples, as well as generally when recording sampling effort for all protocols, eMoF enables greater flexibility to define parameters, as well as the ability to describe the entire sample and treatment protocol through multiple parameters. eMoF also allows you to standardize your terms to a controlled vocabulary.

While vocabularies can be found in the [NERC Vocabulary Server](#), also available through [SeaDataNet](#), we will review specifics on how to find and use controlled vocabularies in Module 5.

We will now move on to Lesson 3 where we will learn how Darwin Core data columns are packaged into core and extension tables, and how these tables are then packaged together.

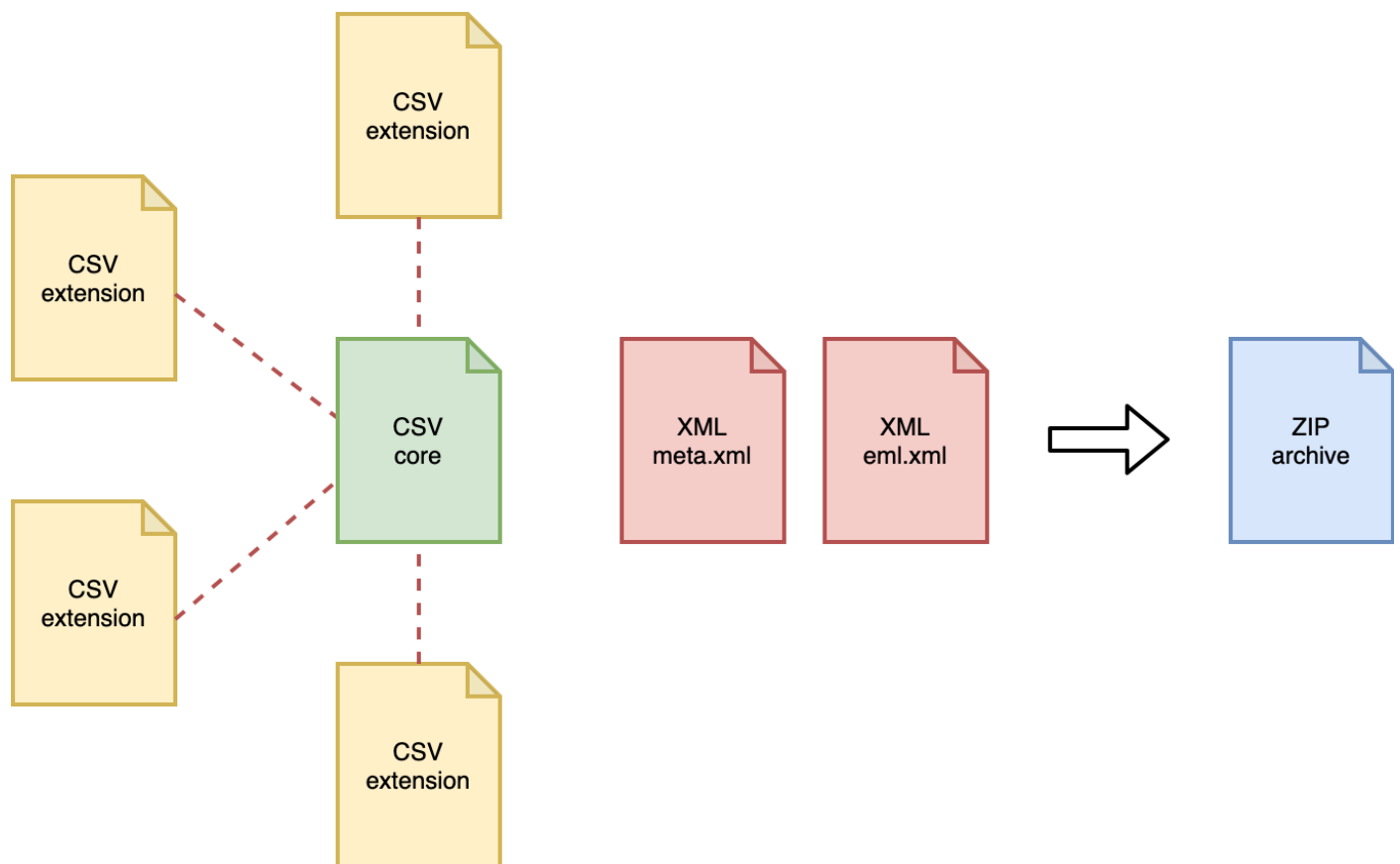
Darwin Core Archive

Now that we understand the basics of Darwin Core terms, let's look at how to standardize the structure datasheets, and how they become packaged together.

The Darwin Core Archive (DWC-A) is the standard for packaging and publishing biodiversity data using Darwin Core terms. It is the preferred format for publishing data in OBIS and GBIF. You can read more details about the format in the [Darwin Core text guide](#). Simply, a Darwin Core Archive is a zipped file containing a number of text files, including data tables formatted as CSV. The data tables contain the columns with DwC terms as column names, where each table contains related information, e.g. all columns related to the occurrence, or all columns related to recording the event. We'll learn about the different data table types in this lesson.

The conceptual data model of the Darwin Core Archive is a **star schema** with a single core table, for example, containing occurrence records or event records at the center of the star. Extension tables can optionally be associated with the core table. Note in this star schema it is **not possible to link extension tables to other extension tables** (to form a so-called snowflake schema). There is a one-to-many relationship between the core and extension records, so each core record can have zero or more extension records linked to it, and each extension record must be linked to exactly one core record. Definitions for all core and extension tables can be found [here](#) (but note that more tables than what are currently integrated by OBIS are listed).

Besides data tables, a Darwin Core Archive also contains two XML files: one file which describes the archive and data file structure (*meta.xml*), and one file which contains the dataset's metadata (*eml.xml*). The figure below shows a representation of this star schema.



In OBIS, there are a few data table types you can choose from that will hold your data, including:

- Occurrence core or extension
- Event core
- MeasurementOrFact or extendedMeasurementOrFact (eMoF)
- DNA Derived Data

Darwin Core-Archive data tables

Let's briefly look at the four main data tables in an OBIS Darwin Core-Archive so we have a better understanding of how data might be separated.

Occurrence table

Occurrence data tables are the simplest and describe the biological observations. This table can be a core table or an extension table. Each row will minimally record will the taxon name, its presence or absence, and the type of record (basisOfRecord). Where this table is the core, location information (date, coordinates, location name) can also be included.

Event table

In the current schema, Event data tables will always be a core table, never an extension. This table is meant to provide only information about the events, i.e. the when (date, time, geologic period, etc.) and where (location, coordinates, depth, etc.) a specific sampling event occurred. It is particularly useful to describe situations where hierarchical event sampling has taken place, and when multiple samples are associated with different events (e.g. many species presence per planktonic tows at different sites).

(Extended)MeasurementOrFact table

This table is always an extension and is meant to hold all measurement data, whether biological, abiotic, environmental, etc. It is formatted in a "long" format instead of wide, where each measurement is a new row within the columns measurementType, measurementUnit, and measurementValue. We'll learn about how to populate this table in Module 3.

DNA derived data table

This table is always an extension, specifically an extension to the Occurrence table. In the current schema, when a dataset is based on DNA data, it will always be Occurrence core + DNA derived data extension. This extension table is meant to hold information that describes the DNA data, e.g. DNA sequences, methodology, primers, etc. More information about this table will be covered in Module 4.

Now we know more about the different data tables, but how do we know when to use which core type (Occurrence vs Event), or which extension to use? For now, know that you must have **one** core data table, and either **zero, one, or more** extensions. We will cover specifics on when to use which core in Module 2. Let's move on to an innovation by OBIS that builds on the star schema on the previous page.

Breaking the star schema: ENV-DATA approach

We learned before that extension files cannot be linked to one another in the traditional Darwin Core Archive star schema. However, data collected as part of marine biological research often include measurements of habitat features (such as physical and chemical parameters of the environment), biotic and biometric measurements (such as body size, abundance, biomass), as well as details regarding the nature of the sampling or observation methods, equipment, and sampling effort. Linking all this information can be difficult when limited to only the star schema.

In the past, OBIS relied solely on the [Occurrence Core](#), and additional measurements were added in a structured format (e.g. JSON) to the Darwin Core term `dynamicProperties` inside the occurrence records. This approach had *significant* downsides: the format is difficult to construct and deconstruct, there is no standardization of terms, and attributes that are shared by multiple records (think sampling methodology) have to be repeated many times. The formatting problem can be addressed by moving measurements to a [MeasurementOrFacts](#) extension table, but that doesn't solve the redundancy and standardization problems.

With the release and adoption of a new core type ([Event Core](#)), it became possible to associate measurements with nested events (such as cruises, stations, and samples), but the restrictive star schema of Darwin Core archive prohibited associating measurements with the event records in the Event core *as well as* with the occurrence records in the Occurrence extension. For this reason, an **extended version of the existing MeasurementOrFact** extension was created.

Let's learn how this extension breaks the star schema to allow links between extensions.

Introducing the ExtendedMeasurementOrFact Extension (eMoF)

As part of the IODE pilot project [Expanding OBIS with environmental data OBIS-ENV-DATA](#), OBIS introduced a custom [ExtendedMeasurementOrFact](#) (eMoF) extension, which extends the existing [MeasurementOrFact](#) extension with 4 new terms:

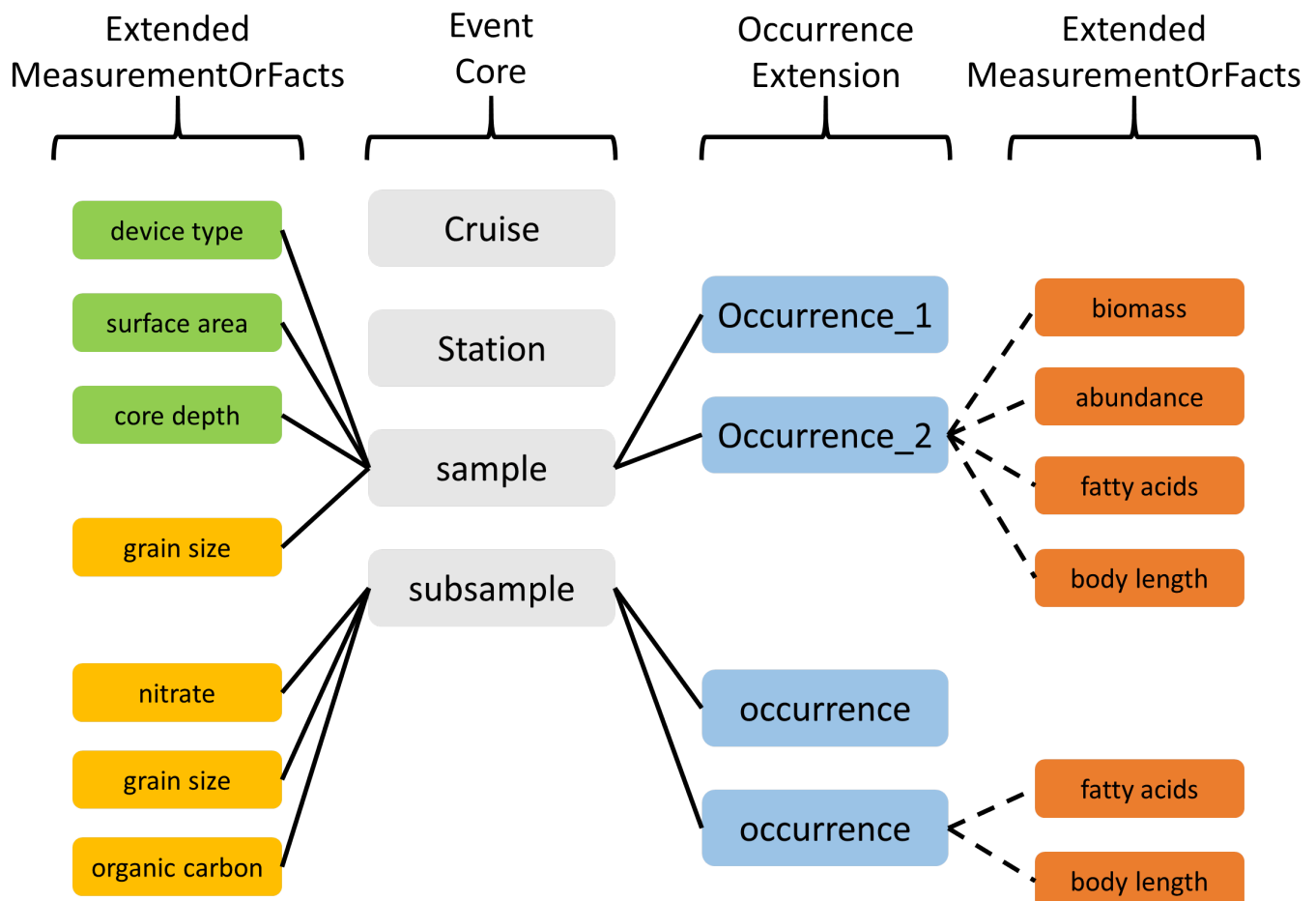
- `occurrenceID`
- `measurementTypeID`
- `measurementValueID`
- `measurementUnitID`

The last three are identifier fields which can be populated with controlled vocabulary, more details on how to select vocabulary is covered in Module 5.

The `occurrenceID` term is used to circumvent the limitations of the star schema, and link measurement records in the eMoF extension to occurrence records in the Occurrence extension. Note that in order to comply with the Darwin Core Archive standard, these records still **need to link to a specific event listed in the Event core table** as well. Thanks to this term we can now store a variety of measurements and facts linked to either events or occurrences! This could include:

- organism quantifications (e.g. counts, abundance, biomass, % live cover, etc.)
- species biometrics (e.g. body length, weight, etc.)
- facts documenting a specimen (e.g. living/dead, behaviour, invasiveness, etc.)
- abiotic measurements (e.g. temperature, salinity, oxygen, sediment grain size, habitat features)
- facts documenting the sampling activity (e.g. sampling device, sampled area, sampled volume, sieve mesh size).

The diagram below demonstrates these links. There is the Event core which has information related to the different types of events, then linked to the events are facts about the sampling methods (device type, etc.) which go in the eMoF, and information regarding occurrences in the Occurrence extension. We can see that facts or measurements related to the occurrences are also placed in the eMoF, but linked to the Occurrence via `occurrenceID`.



This structure is based on relational databases. If you are unfamiliar or uncertain about how such database structures work, we review this

concept on the next page. If you are already familiar with this, you can skip ahead to learn how these structures reduce redundancy in data that is published.

Relational databases

If you are not familiar with relational databases, it can be difficult to understand the framework underlying OBIS. This section will help you understand relational databases and how they relate to OBIS, to the data you will format for OBIS, and to the data you may download from OBIS.

Why do we use relational databases in the first place? You are probably familiar with flat databases which contain all data in one table - this is likely how your own data are formatted. Relational databases instead consist of multiple data tables that each contain *related* information (inter-connected). When all this information is presented in one table, the table becomes larger, very complicated, and the likelihood of data duplication increases. Relational databases seek to simplify complexities and reduce redundancy by allowing information to be self-contained, but linked to each other.

You can think of a relational database as separate Excel sheets or data tables that are connected to each other by at least one term (commonly referenced as "ID"). OBIS relational databases are organized around one "core" table, whereas others are "extensions" of this core. There is always a *relationship* linking core and extension tables.

Let's review core and extension tables and how we use them for OBIS.

OBIS core tables contain information that is applicable to **all** extension tables, and extension tables contain *more information* about the records within the Core table. Each table, whether core or extension, contains records and attributes. Each row is a record (e.g., a sampling event, a species' occurrence), whereas each column is an attribute (e.g., a date, a measurement).

Records are linked between tables by the use of *identifiers* (e.g., *occurrenceID* or *eventID* in OBIS). A description of measurements pertaining to a record in an Extension table will have the same identifier as the record it is describing in the Core table. By using identifiers to link records, we reduce data repetition.

In the Darwin Core format that OBIS uses, **the core table is either Event or Occurrence**, and datasets can have one, none, or more extension tables. Further explanation of data formatting in OBIS is covered in Module 2 and 3 as well as the Data Formatting section of the [OBIS Manual](#).

Relational database example

We will look at a simple relational database used by a fictional country that tracks student performance in three different courses between three schools. Rather than trying to contain information about each school, course, and student performance in one place, this information is split into three separate tables. We see that the pink table below gives us information about each school - its name, and the district it belongs to. Each school also has a schoolID, an identifier linking to the blue table where we can see student performance (course mean) in each course, the class size, and year. You will notice that the course mean and class size are bundled under columns called measurementType and measurementValue, this is to simplify all measurements into a long format as is done in the eMoF. We'll see more about this in Module 3. This eMoF structure is integral to reducing repeated data, especially when one dataset has reoccurring information. Finally, we see that the courseID in the blue table links to the yellow one with the courseID identifier, giving us information about each course.

A fourth table could easily be created to track total school population size through time. In contrast, if this information was only presented in the pink Schools in Country table, the school information would be duplicated as you add rows for each year. In this way, you can easily see how useful relational databases are. Of course, this is a simplified example that does not implement the star schema we learned about, but it demonstrates how related tables can be linked by identifiers to reduce table complexity and data replication.

Schools in Country		
schoolID	districtName	schoolName
A1	Crystal County	Waterfalls Conservatory
A2	Crystal County	Valley View Elementary
A3	Upper Lake	Oak Ridge

Student Performance				
schoolID	courseID	measurementType	measurementValue	year
A1	C1	classSize	31	2008
A1	C1	mean	73.8	2008
A2	C2	classSize	26	2008
A2	C2	mean	64.3	2008
A3	C3	classSize	23	2008
A3	C3	mean	78.2	2008

Courses		
courseID	courseName	updated
C1	Biology	2004
C2	Math	2003
C3	Social Studies	2006

Let's continue on with how such structuring reduces redundancy in OBIS datasets.

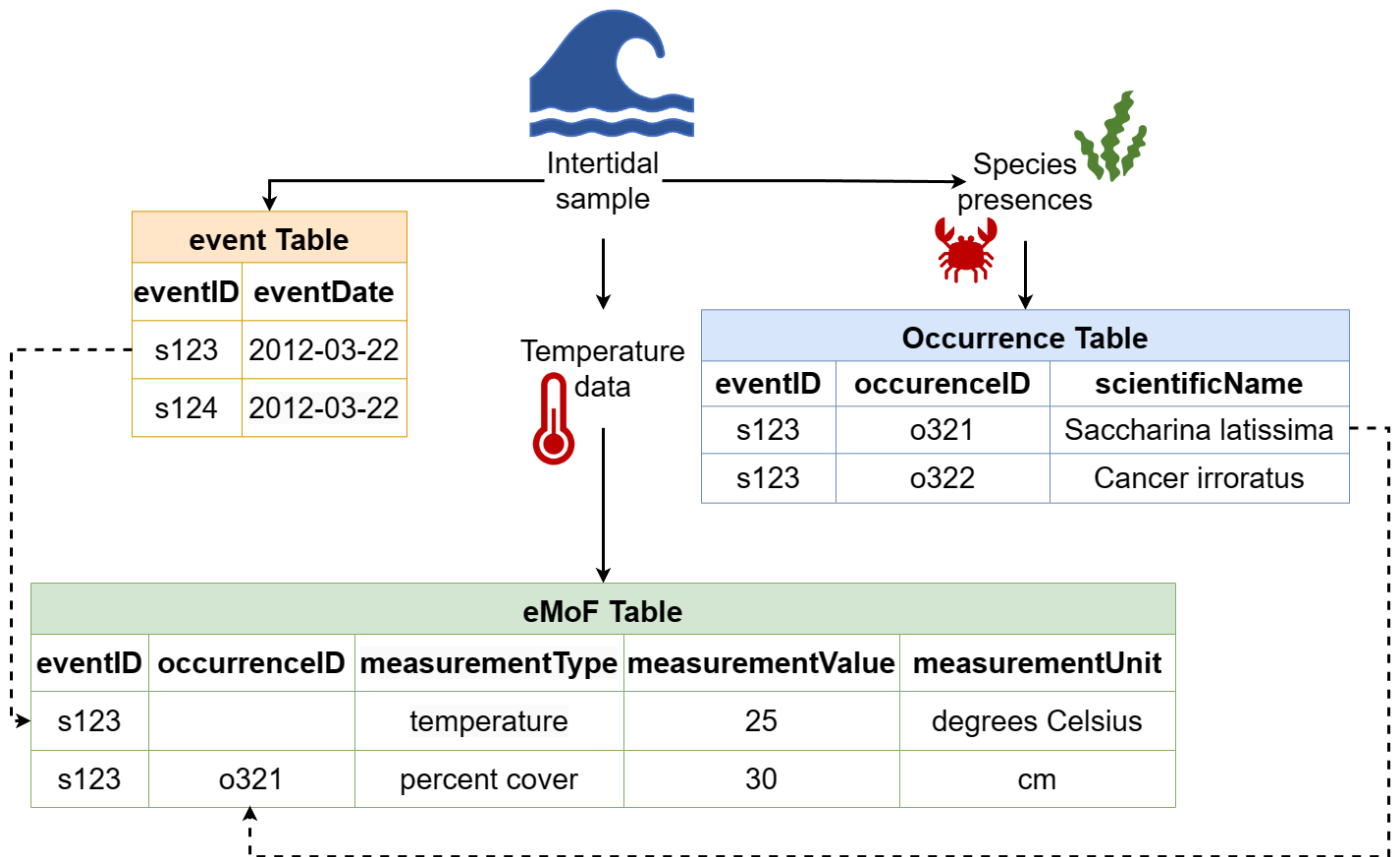
Reducing Redundancy

Using the OBIS data structure (ENV-DATA) allows you to avoid redundancy and data duplication within your dataset. We can limit the repetition of data by applying the ENV-DATA approach, which delineates relationships between the core table and extension tables.

For example, let us consider the dates of a ship cruise that carried out a series of bottom trawls. The sampling information (e.g., date range, equipment used, etc.) for each species collected in these trawls is the same and corresponds to the log of the sampling station where all the organisms were recorded. We know we are dealing with unique sampling events because for each new sampling station, you are going to have a totally new set of sampling information, and thus, occurrences will be organized around an Event core. So, our Event core table will contain all information related to the sampling events (e.g., date, geographical coordinates, habitat, depth, etc.), and the taxa information will be placed in the Occurrence table.

Then, information pertaining to each collected specimen/taxon (e.g., abundance, biomass, sampling methods, etc.) will be placed in an extension, another spreadsheet called the (Extended)MeasurementOrFact table. Here, measurements for each specimen/taxon and sample will occur on a separate record. All these records (rows in the spreadsheet) will be linked to a specific sampling event or occurrence by an identifier - the eventID (to link to an Event) or occurrenceID (to link to an Occurrence). If we were to put this data in one file, the fields related to date and location (e.g., eventDate, decimalLongitude, decimalLatitude, etc.) would be repeated for each species.

Let's consider another example. If you took one sea temperature measurement where you also took your intertidal sample, each organism found in that sample would have the **same** temperature measurement. By linking such measurements to the Event table instead of in the Occurrence table, we are able to reduce the amount of data being repeated.



An advantage of structuring data this way is that if any mistakes are made, you only need to correct it once! So you can see that using relational Event structures (when applicable) in combination with extension files can really simplify and reduce the number of times data are repeated.

Caveat: we would like to note that in some cases, data duplication may occur due to the star schema structure. For example, when publishing DNA-derived data, Occurrence core will have to be used, which necessitates the repetition of event data for each occurrence record.

The following resources and references can provide additional information:

- [Darwin Core Quick Reference Guide](#)
- [Darwin Core text guide](#)
- De Pooter et al. 2017. Toward a new data standard for combined marine biological and environmental datasets - expanding OBIS beyond species occurrences. Biodiversity Data Journal 5: e10989. hdl.handle.net/10.3897/BDJ.5.e10989
- Duncan et al. (2021). A standard approach to structuring classified habitat data using the Darwin Core Extended Measurement or Fact Extension. EMODnet report. <https://www.emodnet-seabedhabitats.eu/resources/documents-and-outreach/#h3298bcd0a15741a8a0ac1c8b4576f7c5> (note you must refine search to Technical Reports from 2021 to identify this report as it does not have an individual link)

You've completed this module! Please complete Exercise 1-1 and Quiz 1, and then proceed to the next module.

Module 2: Introduction to data formatting

Site: [OceanTeacher Global Academy](#)

Course: Contributing and publishing datasets to OBIS (self-paced)

Book: Module 2: Introduction to data formatting

Table of contents

Module 2

Lesson 1: Determining dataset structure

- When to use Occurrence Core
- When to use Event Core
- Lesson summary

Lesson 2: Understanding and constructing Identifiers

- Construct eventID
- Construct occurrenceID
- Lesson Summary

Lesson 3: Taxon match

- Taxon Matching Workflow
- Matching with WoRMS
- Match with other registers
- Non-marine taxa
- Watch videos
- Tools for Taxon matching

Lesson 4: Addressing common formatting issues

- Dates and times
- Converting coordinates
- Missing required fields
- Lesson Summary
- End of Module

Introduction

In this module, you will learn about the necessary first steps required to format data tables that will adhere to Darwin Core standards. You will learn how to match taxonomic names to an authoritative register, how to construct identifiers discussed in Module 1 (occurrenceID, eventID), and how to standardize data (e.g., dates, coordinates). You will also learn how to address missing data for required fields.

Learning Outcomes

After successful completion of this module, you should be able to:

- Prepare a list of species/taxa for input into WoRMS, resolve ambiguous or non-matches from WoRMS taxon match, obtain LSIDs from WoRMS Taxon Tool, attach LSIDs to own dataset
- Identify hierarchical elements used to construct eventID and put them together to create unique eventIDs
- Construct unique occurrenceIDs
- Format event dates to ISO 8601 standards including time zones and time ranges
- Convert degrees/minutes/seconds coordinates or UTM into decimal degrees

How to Proceed

To succeed in this Module, you need to successfully complete the following lessons and exercises:

- Lesson 1: Determining dataset structure
- Lesson 2: Understanding and constructing Identifiers
- Lesson 3: Taxon match
- Lesson 4: Addressing common formatting issues
- [Exercise 2-1: Dataset structures](#)
- [Exercise 2-2: Construct eventID](#)
- [Exercise 2-3: WoRMS Taxon names matching](#)

as well successfully complete Quiz 2 with a score of $\geq 80\%$

- [Quiz 2](#)

Determine your dataset structure

Introduction

Because formatting data in a standardized way can be challenging, this module starts with an overview of a dataset structure. Deciding on your dataset structure is one of the first steps towards getting your data ready for publishing. We've already been introduced to the different Darwin Core data table types, and it is important to determine which structure (core + extension table(s)) best suits your dataset next. Then, once you have decided on the dataset structure, you can continue formatting your data.

As a review, there are two possible types of core data tables for OBIS: Occurrence core and Event core. Let's look at each one, and when either should be chosen as your core data table.

When to use Occurrence Core

Occurrence Core datasets describe **observations** and **specimen records** and cover instances when:

- **No information** on how the data was sampled or how samples were processed is available
- No abiotic measurements are taken or provided
- You have eDNA and DNA-derived data
- Biological measurements are made on **individual specimens** (each specimen is a single occurrence record)

Occurrence Core is also often the preferred structure for museum collections, citations of occurrences from literature, and sampling activities.

Datasets formatted in Occurrence Core can use the eMoF Extension for when you have biotic measurements or facts about your specimen. The DNA derived data extension can, and should, be used to link to DNA sequences and other bioinformatic data to the Occurrence core. The identifier that links Occurrence Core to the extension(s) is the **occurrenceID**.

Next, let's learn when Event core should be used.

When to use Event Core

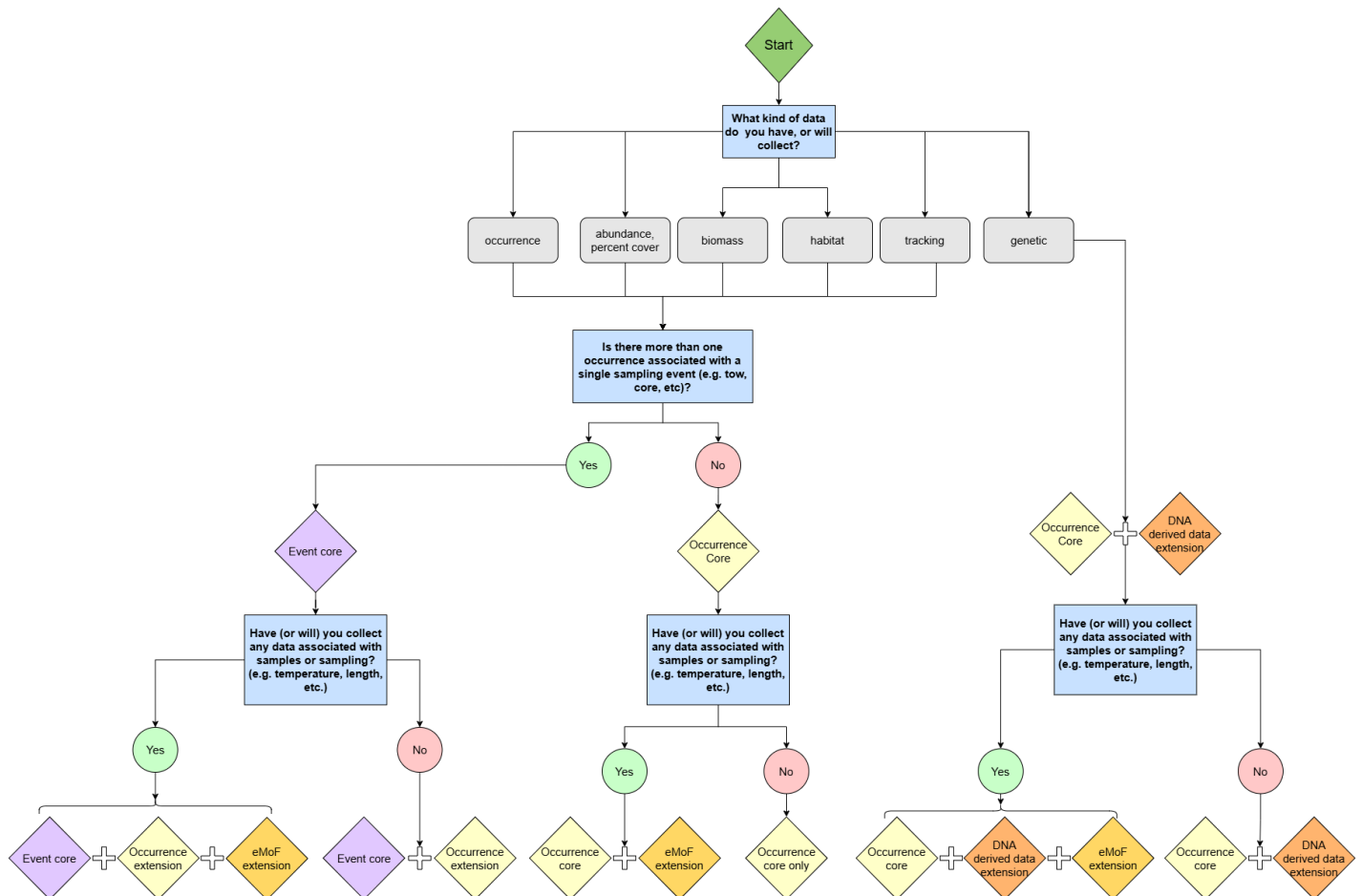
Event Core describes **when** and **where** a specific sampling event happened and contains information such as location and date. Event Core is often used to organize your data tables when there are more than one sampling occasion and/or location, and different occurrences linked to each sampling. This organization follows the rationale of most ecological studies and typical marine sampling design. It covers:

- When specific details are known about **how** a biological sample was taken and processed. These details can then be defined in the eMoF Extension with the [Q01 vocabulary](#)
- When the dataset contains abiotic measurements, or other/biological measurements which are **related to an entire sample** (not a single specimen). For example a biomass measurement for an entire sample, not each species within the sample

Event Core can be used in combination with the Occurrence and eMoF extensions. The identifier that links Event Core to the extension is the `eventID`. `parentEventID` can also be used to give information on hierarchical sampling. `occurrenceID` can be used in datasets with Event Core in order to link information between the Occurrence extension and the eMoF extension.

As established in Module 1, OBIS relies on datasets being formatted with relations to one another. Simply, it means that data tables are connected to each other by identifiers and each data table contains a set of data about a different measurement (or fact) characterizing either the occurrence or the sampling (observation/occurrence). The ENV-DATA approach that OBIS implements means your dataset will have a core table and (optionally) extension table(s). In this lesson we learned more about the core tables integrated by OBIS (Occurrence or Event), and the associated potential extension tables(s) (eMoF, Occurrence, DNA) that are linked to core tables by the use of identifying ID codes. These codes could be either *eventID* or *occurrenceID*. We will learn about these important identifiers, and how to construct them, in the next lesson.

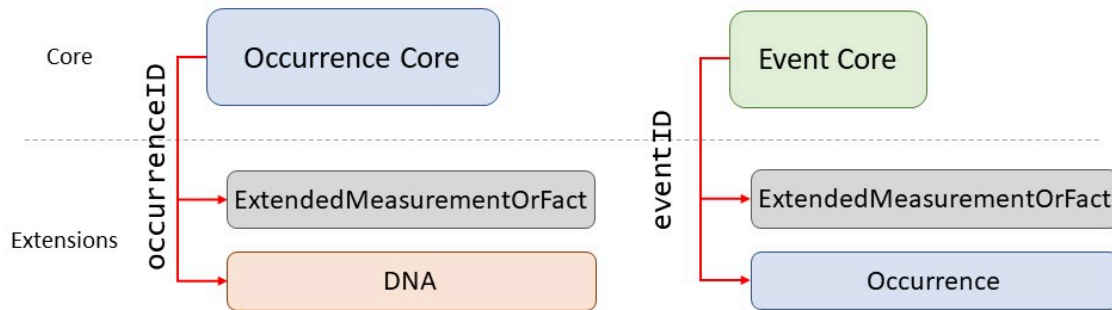
To practice your understanding on when to select Occurrence vs Event core, complete Exercise 2-1. See the flowchart below for an additional summary of this lesson and/or to help you in the Exercise.



Introduction to Identifiers

As we know, there are two important identifiers that link your data tables: `occurrenceID` and `eventID`.

If your core file is based on occurrences (e.g., a record of one or more taxa specimens), then any extensions are linked with `occurrenceID`. If your core file is based on events (e.g., a sampling event, cruise, observation, etc.), then the linking identifier is `eventID`. In the Core tables, identifiers are always unique, which means, they do not repeat and each line has a different identifier. On the other hand, multiple records in an extension file can have the same identifier which will link them to the same event or occurrence record (depending on which is the Core). The different linking identifiers are shown in the figure below.



Let's learn how to construct each identifier, beginning with `eventID`.

Using a unique identifier for each physical sample or subsample in your dataset taken at each location and time is highly recommended to ensure sample traceability and data provenance. `eventID` is an identifier for an individual sampling or observation event, whereas `parentEventID` is an identifier for a parent event, which is composed of one or more linked (child) events (eventIDs).

`eventID` can be used to distinguish between replicated samples and/or sub-samples. It is important to make sure each replicate sample receives a unique `eventID`, which could be based on an already existing unique `sample ID` in your dataset. Sample ID can also be recorded in `materialSampleID`, as OBIS does not need to have separate `eventIDs` and `materialSampleIDs`. Rather OBIS can treat these two terms as equivalent. But you must still fill in the `eventID` field if you want to use `materialSampleID`, as OBIS only uses `eventID` and `parentEventID` for structuring datasets, *not* sample ID. This does not prevent you from using the field if you would like to!

`eventIDs` can be machine-generated or human-generated. There are pros and cons to each, but we focus on human-generated IDs in this course.

If you do not already have a sample ID, creating a unique `eventID` for your samplings can be as straightforward as combining different fields from your data. When constructing `eventIDs`, ask yourself what is the main information about a sampling event that helps you identify it? For instance, it is helpful when we know the location, date, project, habitat, type of subsample, etc. You can build your `eventID` code based on this information and ensure IDs will not be repeated for different events (e.g., will result in a unique identifier). When initially constructing `eventIDs`, there may be rows of data that come from the same event, in this case it's okay for them to have the same `eventID`. We will learn how to deal with this when we separate occurrences and event information in Module 3.

Note You should consider carefully what combination of fields will generate a **unique** `eventID` for each unique event. Combinations including project, date, time, location, and depth are common elements to help generate such unique codes.

Including the `event type` can also be useful for datasets with hierarchical sampling methods (e.g., samples taken from a station within a cruise). Repeating the `parentEventID` (using `:` or `_` as a delimiter) can also make the structure of the dataset easier to understand. Nesting event information in this way also allows you to reduce redundancy while still providing information relevant to each level of sampling.

Broadly, an `eventID` can take the form of `[parentEventID]_[sample type]_[sample ID]`

Thus to construct a unique `eventID` for parent and child events, you can join relevant sampling information. Possible configurations (with examples) could include:

- Project_cruise_station_date_sample
 - STAR_arcticsea_st3520_1989-04-04_s01
- Project_habitat_Genus_species_year_sampletype_samplenumber
 - BEE_seamount_Genus_species_2013_cruise_s123
- Institution_year_location_samplemethod_sample
 - Concordia_2003_Coast_Station1_seine_s01
 - Concordia_2003_Coast_Station1_trap_s01

These examples are not exhaustive and other similarly structured variations that fit your data are acceptable. Consider also including year within your `eventIDs` to ensure codes remain globally unique in subsequent years, which is particularly useful if your sampling protocol is repeated temporally.

Information related to your sampling events can be assigned to the highest relevant event level in order to avoid repetition of information because parent event information is passed to child events. For example, if all samples taken from a station occurred at the same depth, this information can be listed once. Variation between samples (e.g., exact time or coordinates) can also be easily reflected for each event. See the table below for a simple demonstration. **However for datasets that will also be harvested by GBIF**, we recommend populating both parent and child events because child events in GBIF do not currently inherit information from the parent event like they do in OBIS.

Examples

	A	B	C	D	E
1	<code>eventID</code>	<code>parentEventID</code>	<code>eventDate</code>	<code>eventRemarks</code>	<code>maximumDepthInMeters</code>
2	<code>cruise_1</code>			<code>cruise</code>	
3	<code>cruise_1:station_1</code>	<code>cruise_1</code>		<code>station</code>	15
4	<code>cruise_1:station_1:core_1</code>	<code>cruise_1:station_1</code>	2011-03-06T08:35	<code>sample</code>	
5	<code>cruise_1:station_1:core_2</code>	<code>cruise_1:station_1</code>	2011-03-06T08:52	<code>sample</code>	
6	<code>cruise_1:station_1:core_1:subsample_1</code>	<code>cruise_1:station_1:core_1</code>		<code>subsample</code>	

Watch a video

Watch the video below for a demonstration on how to construct eventIDs (available at <https://youtu.be/Upt6LPJ0Bn8>)

Similar to **eventID**, **occurrenceID** is an identifier to distinguish between occurrence records where each **occurrenceID** must be unique. Because **occurrenceID** is a required term, you may have to construct a persistent and globally unique identifier for each of your data records if none already exists (e.g., if records were not labeled with unique identifiers before, such as during sample processing or image/sensor detection).

There are no standardized guidelines yet on designing the persistence of this ID, the level of uniqueness (from within a dataset to globally in OBIS), and the precise algorithm and format for generating the ID. But in the absence of a persistent globally unique identifier, one can be constructed by combining the **eventID** with **institutionCode**, the **collectionCode** and/or the **catalogNumber** (or autonumber in the absence of a **catalogNumber**). This will be similar to how **eventID** is constructed. You may also follow [Life Science Identifiers](#) guidelines. The inclusion of **occurrenceID** is also a required term for OBIS datasets.

An important consideration for museum specimens: there is the possibility that the institution a specimen is housed at may change. Therefore you may consider omitting institution identifiers within an **occurrenceID**, because **occurrenceID should not change over time**.

Example

Data from [Algal community on the pneumatophores of mangrove trees of Gazi Bay in July and August 1987](#).

occurrenceID	basisOfRecord	institutionCode	collectionCode	catalogNumber
Ugent_Vegetation_Gazi_Bay(Kenya)1987_7553	HumanObservation	Ugent	Vegetation_Gazi_Bay(Kenya)1987	Ugent_Vegetation_Gazi_Bay(
Ugent_Vegetation_Gazi_Bay(Kenya)1987_7554	HumanObservation	Ugent	Vegetation_Gazi_Bay(Kenya)1987	Ugent_Vegetation_Gazi_Bay(
Ugent_Vegetation_Gazi_Bay(Kenya)1987_7555	HumanObservation	Ugent	Vegetation_Gazi_Bay(Kenya)1987	Ugent_Vegetation_Gazi_Bay(

Watch a video

Watch the video below for an example of how to construct an **occurrenceID**, which starts at 1:52 and can also be found at https://youtu.be/G_AmAmS7ILc?t=112

Let's summarize what we have learned about identifiers:

- Identifiers link information between data tables
- Identifiers should be unique and persistent, they should not change over time
- You can construct identifiers from information in your dataset

Now that we have a better understanding of identifiers, practice constructing eventIDs for an example dataset in Exercise 2-2. Then, we will learn about another type of identifier in the next lesson: taxonomic identifiers.

Introduction to Taxonomic Identifiers

In the previous lesson we learned about the identifiers that link our data tables: `occurrenceID` and `eventID`. In this lesson we will learn about the taxonomic identifier you should use to fill the Darwin Core field `scientificNameID`.

OBIS requires all your specimens to be classified and matched against an authoritative taxonomic register. This effectively attaches unique stable identifiers (and digitally traceable) to each of your species. Meaning, if a taxonomic ranking or a species name changes in the future, there will be no question as to which species your dataset is actually referring to. Matching to registers also helps to avoid misspelled or unused terms.

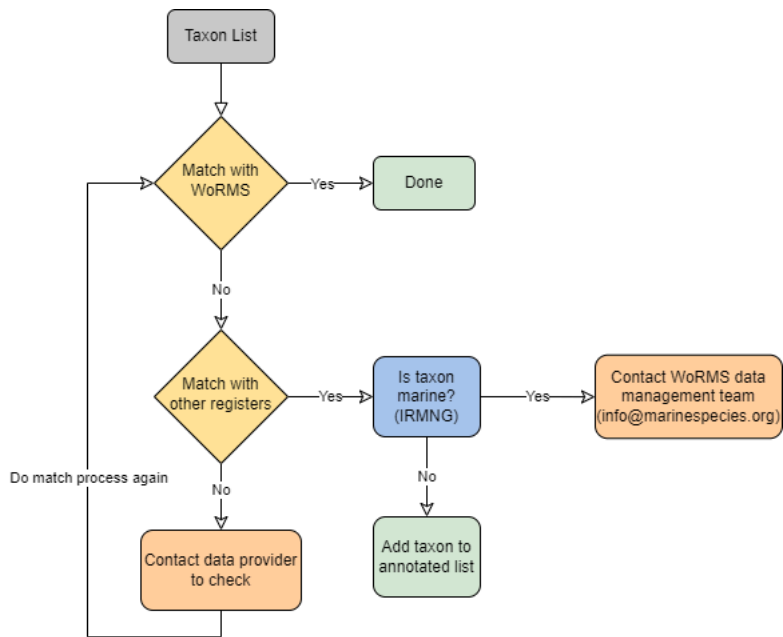
OBIS currently accepts identifiers from **three** authoritative lists:

- [World Register Marine Species \(WoRMS\)](#) LSIDs
- [Integrated Taxonomic Information System \(ITIS\)](#) TSNs
- [Barcode of Life Data Systems \(BOLD\)](#) and [NCBI](#) identifiers

The identifiers (LSID, TSN, ID) from these registers will be used to populate the `scientificNameID` field. OBIS can accept other LSIDs besides WoRMS, as long as they are mapped in WoRMS. If you would like to include multiple identifiers, please use a concatenated list where each register is clearly identified (e.g. [urn:lsid:itlis.gov:itlis_tsn:12345](#), NCBI:12345, BOLD:12345). You can also use the [Interim Register of Marine and Nonmarine Genera \(IRMNG\)](#) to distinguish marine genera from freshwater genera.

You should **prioritize using LSIDs** because they are unique identifiers that indicate the authority the ID comes from. We recommend **obtaining LSIDs from WoRMS** because this is the taxonomic backbone that OBIS relies on, and it is built on marine systems as well as linked to the other taxonomic authoritative lists. Let's take a look at how to go about matching a list of scientific names to a register like WoRMS.

To match the scientific names in your dataset, you should follow this Name Matching workflow:



This workflow can be broken into two main steps 1) Match with WoRMS, 2) Match with other registers.

Let's start by taking a look at how to match to WoRMS by watching the video below on how to use the WoRMS Taxon Match tool (available <https://youtu.be/jj8nIMlg-cY>):

Now let's review the steps involved in matching to WoRMS.

The procedure for matching to WoRMS and then attaching successful matches back to your data can be simplified to:

1. Prepare a file (.csv, .txt, .xlsx, etc.) with the list of your specimens/taxa
2. Upload the file to WoRMS taxon match tool
 - o Check relevant boxes
3. Review returned file
4. Resolve any ambiguous matches
5. Download file and identify data to include in your Occurrence data table for OBIS
 - o LSIDs, taxonomic fields, etc.
6. Attach LSIDs back to your data using e.g.:
 - o R (merge)
 - o Excel (vlookup)

The **taxon match tool** of the World Register of Marine Species (WoRMS) is an automatic way to match and download the taxonomic information about your occurrence records, without having to look for each name in the site. It is available at <http://www.marinespecies.org/aphia.php?p=match>. The WoRMS taxon match will compare your taxon list to the taxa available in WoRMS.

This taxon match takes into account exact matches and fuzzy matches. Fuzzy matches include possible spelling variations of a name available in WoRMS. WoRMS also identifies ambiguous matches, indicating that several potential matching options are available (e.g. homonyms). You can check these ambiguous matches and select the correct one, based on e.g., the general group information (a sponge dataset) or the authority. If this would be impossible with the available information (e.g., missing authority or very diverse dataset), then you need to contact the data provider for clarification.

For performance reasons, the limit is set to 1,500 rows for the taxon match tool. Larger files can be sent to info@marinespecies.org and will be returned as quickly as possible. In case you have recorded a taxon that is not registered in WoRMS (e.g., newly discovered species), you should contact them so the database can be updated.

After matching, the tool will return you a file with the AphiaIDs, LSIDs, valid names, authorities, classification, and any other output you have selected.

Reminder: The WoRMS LSID is used to populate `DwC:scientificNameID`.

A complete online manual is available at <http://www.marinespecies.org/tutorial/taxonmatch.php>.

If you do not find a match with WoRMS and there are no spelling or formatting errors, you should next check other registers.

The [LifeWatch taxon match](#) compares your taxon list to multiple taxonomic standards. Matching with multiple registers gives an indication of the correct spelling of a name, regardless of its environment. If a name would not appear in any of the registers, this could indicate a mistake in the scientific name and the name should go back to the provider for additional checking/verification. You will need to create a login to use the LifeWatch taxon match.

Contrary to the WoRMS taxon match, when several matching options are available, the LifeWatch taxon match only mentions “no exact match found, multiple possibilities” instead of listing the available options. If multiple options are available, these should be looked up and matched manually.

Currently, the LifeWatch web service matches the scientific names with the following taxonomic registers:

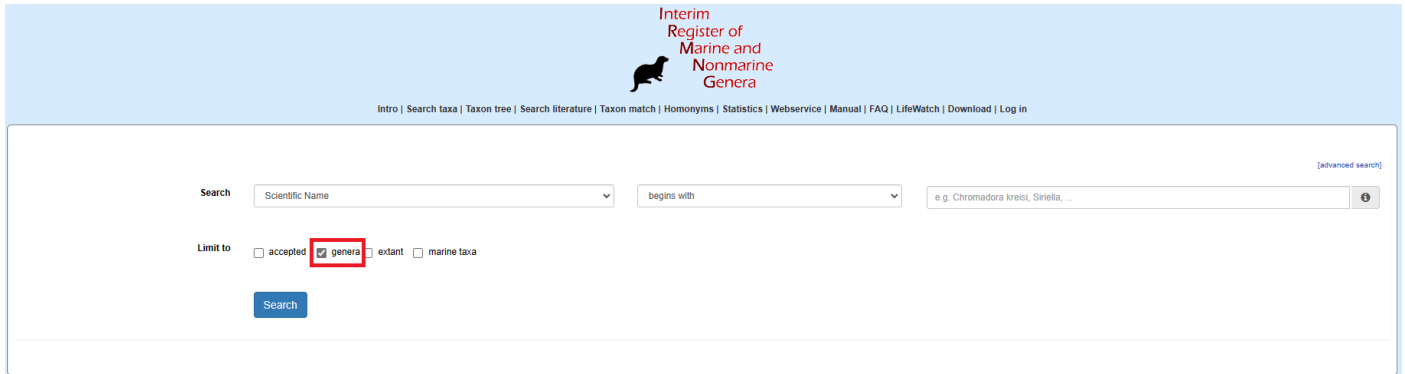
- [World Register of Marine Species - WoRMS](#)
- [Catalogue of Life - COL](#)
- [Integrated Taxonomic Information System - ITIS](#)
- [Pan-European Species-directories Infrastructure - PESI](#)
- [Index Fungorum - IF](#)
- [International Plant Names Index - IPNI](#)
- [Global Names Index - GNI](#)
- [Paleobiology Database - PaleoDB](#)

Keep in mind that it can take some time for the service to process your file, which is why we recommend using the WoRMS taxon match service first.

After cross-referencing with LifeWatch, if you still have non matching species, the next step is to **confirm if the taxa are marine**.

To check whether a taxon is marine or not, you can cross-reference with the Interim Register of Marine and Non-marine Genera (IRMNG). These matching services are available through <http://www.irmng.org/> (as well as through the [LifeWatch taxon match](#)).

This service allows you to search for a genus (or other taxonomic rank when you uncheck the “genera” box, see below) to check if it is known to be marine, brackish, freshwater, or terrestrial. You can find this information in the row labeled “Environment”. If the taxon is marine but is not listed in WoRMS, you may have to contact the WoRMS data management team (info@marinespecies.org) to have the taxon added to the WoRMS register. You will have to provide supporting information confirming taxonomic and marine status.



The screenshot shows the search interface of the Interim Register of Marine and Non-marine Genera (IRMNG). The header features the logo and name of the register. Below the header is a navigation menu with links: Intro | Search taxa | Taxon tree | Search literature | Taxon match | Homonyms | Statistics | Webservice | Manual | FAQ | LifeWatch | Download | Log in. The main search area includes a search type dropdown set to "Scientific Name", a "begins with" dropdown, and a search input field containing "e.g. Chromadora kreisi, Siniella, ...". There is a "Limit to" section with checkboxes for "accepted", "genera" (which is checked and highlighted with a red box), "extant", and "marine taxa". A "Search" button is located below the search fields. A "[advanced search]" link is visible in the top right corner of the search area.

The video below demonstrates several examples of how to resolve fuzzy or ambiguous matches from WoRMS, including how to deal with non-marine species (available at <https://youtu.be/yZKwtr14JVM>):

17 Obtaining WoRMS identifiers part 2 - How to resolve ambiguous m...



Play Video

This video reviews an example of species synonyms (available at <https://youtu.be/-vIkS8U7Crk>):

18 Obtaining WoRMS identifiers part 3 - How to resolve species nam...



Play Video

Let's next look at a summary for all the tools available to you to complete taxon matching.

There are a number of tools available to help you match a list of names to WoRMS. We have summarized these tools the table below, as well as advantages and disadvantages for each.

Tool	Advantage	Disadvantage
WoRMS taxon match	<ul style="list-style-type: none">• Accessible online• Does not require coding knowledge	<ul style="list-style-type: none">• Requires rematch information back to your data• The limit is set to 1,500 rows
R package obistools::match_taxa	<ul style="list-style-type: none">• Produces same output as WoRMS taxon match• Already in R so easier to merge back with data• Allow user to match taxa interactively• Do not need to loop over data	<ul style="list-style-type: none">• Outputs a tibble for taxa names specified• Requires knowledge of R
R package worrms::wm_records_taxamatch	<ul style="list-style-type: none">• Outputs all taxonomic information from WoRMS (taxonomic hierarchy, rank, authority)• No limit	<ul style="list-style-type: none">• Outputs a tibble for each taxa name specified• Requires knowledge of R
Python pyworms	<ul style="list-style-type: none">• Simple and easy to understand output	<ul style="list-style-type: none">• Requires knowledge of python• Challenging to deal with when multiple matches occur

Practice using the WoRMS Taxon Match tool in Exercise 2-3, and then move to the next lesson where we will learn about standardizing dates, coordinates, and how to resolve missing required fields.

Common formatting issues

In this lesson we review a few common challenges you may encounter during data formatting, including:

- Standardizing dates and times
- Converting coordinates
- Missing required fields

We will look at each topic in preparation for formatting core and extension data tables in the next module. You may find it easier to standardize data formatting before creating your data tables, or you may find it easier to do this after separating data into the relevant data table. For this course, we will discuss standardizing data first. (Although during the data formatting exercises you may notice some fields are not yet standardized, this is intentional).

The date and time at which an event took place or an occurrence was recorded go in `eventDate`. This field uses the [ISO 8601 standard](#) and we recommend using the extended ISO 8601 format with hyphens. Note that **all dates in OBIS become translated to UTC** during the quality control process implemented by OBIS. Formatting your dates correctly ensures there will be no errors during this process.

ISO 8601 dates can represent moments in time at different resolutions, as well as time intervals, which use `/` as a separator. Date and times are separated by `T`. Timezones can be indicated at the end by using `+` or `-` the number of hours offset from UTC. If no timezone is indicated, then the time is assumed to be local time. When a date/time is recorded in UTC, a `Z` should be added at the end. Times must also be written in the 24-hour clock system. If you do not know the time, you do not have to provide it. Please **do not indicate unknown times as "00:00"** as this indicates midnight.

Not every piece of time information is necessary, but a generalization of how to format dates and times looks like:

```
YYYY-MM-DDT[hh]:[mm]:[ss][+/-XX OR Z]
```

Some specific examples of acceptable ISO 8601 dates are:

Dates:

- 1948-09-13
- 1993-01/02
- 1993-01
- 1993

Dates with Specific Times:

- 1973-02-28T15:25:00
- 2008-04-25T09:53

Dates with Time Zones:

- 2005-08-31T12:11+12
- 2013-02-16T04:28Z

Date and Time Intervals:

- 1993-01-26T04:39+12/1993-01-26T05:48+12

It is important to note that ISO 8601 ordinal dates (YYYY-DDD) and week dates (YYYY-Www-D), are not supported by OBIS. Additionally, ISO 8601 guidelines for [durations](#) should not be used. Durations for an event (e.g., length of observation) can instead be indicated with the DwC terms [startDayOfYear](#) and [endDayOfYear](#). Durations refer to the actual length of time an event (e.g., occurrence) occurred, whereas intervals indicate the time period during which an event was recorded.

A note about intervals... Take care when entering date intervals as, for example, entering 1960/1975-08-04 indicates that the event or observation started any time in 1960, and ended any time on 1975-08-04. If you know the exact date and time, you should specify that information. This also helps for continuous samplings and time-series integrated datasets.

If you have a mix of dates and times for different aspects of a sampling event, you can embed this information in the Event Core table using hierarchies of date structure. To do this, you can use separate records for events, and specify each event date individually.

It is good practice to place the original event date/time description into the `verbatimEventDate` field. Any modifications you make during data formatting should be recorded in the `eventRemarks` field, and we recommend taking good notes in a Documentation sheet (e.g. an extra tab).

You may watch this video for a demonstration on formatting dates according to ISO 8601 standards (<https://youtu.be/0xJWdIeTxo>)

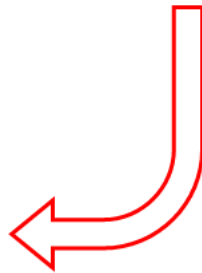
All coordinates provided in the `decimalLatitude` or `decimalLongitude` fields in OBIS must be provided in **decimal degrees**. To convert coordinates from degrees-minutes-seconds into decimal degrees, you can use the OBIS [Coordinate Conversion tool](#). This tool will convert any coordinate or list of coordinates from degrees-minutes-seconds to decimal degrees, even partial coordinates. Simply copy and paste your coordinates into the box provided and click Convert. For example:

Coordinate conversion

Input

```
51°28'38"N 101°16'56"W
51°28'38"N 101°16'56"W
51°28'38"N 101°16'56"W
51o28'38"N 101°16'56"W
51°28'38"n 101°16'56"w
51° 28' 38" N 101° 16' 56" W
51 ° 28 ' 38 " N 101 ° 16 ' 56 " W
51°28'38"N -101°16'56"E
51° N 101° W
51° N
12° N 109° 58' 37" W
```

lon	lat
-101.28222	51.47722
-101.28222	51.47722
-101.28222	51.47722
-101.28222	51.47722
-101.28222	51.47722
-101.28222	51.47722
-101.28222	51.47722
-101.28222	51.47722
-101.00000	51.00000
NA	51.00000
-109.97694	12.00000



Convert

Watch this video for a demonstration on how to use the OBIS coordinate conversion tool (https://youtu.be/E_TkWIUcojw)

As a reminder, there are 8 required fields to publish a dataset to OBIS. To resolve missing fields marked as required by OBIS, follow the guidelines below for each term.

- **occurrenceID** or **eventID**

Create a unique occurrenceID for each of your observations following the guidelines in this module. Recall that these IDs can be generated by combining dates, location names, sampling methods, and/or existing identifiers in the dataset.

- **eventDate**

Ensure your eventDate is specified for each event, formatted according to [ISO 8601 standards](#) (e.g., YYYY-MM-DD) as we learned earlier in this lesson. For any eventDate that is inferred from literature, you should document the original date in the **verbatimEventDate** field.

- **decimalLongitude** and **decimalLatitude**

Make sure all coordinates are converted into decimal degrees, as we learned previously. If you do not have specific coordinate data then you must approximate the coordinates based on locality name. You can use the [Marine Regions gazetteer](#) to search for your region of interest and obtain midpoint coordinates. Guidelines for using this tool and for dealing with uncertain geolocations can be found on the [OBIS manual](#), and we will learn more about this in Module 6. You will need to document georeferencing processes in the **georeferenceRemarks** field if you estimate coordinates.

- **scientificName**

This field should contain only the **originally documented** scientific name down to the lowest possible taxon rank, even if it is a current synonym. Class or even Kingdom levels are accepted if more specific taxonomic levels are unknown. Comments about misspellings, etc. can be documented in the **taxonRemarks** field. You may encounter challenges filling this field if the species name is based on description or if its taxonomy was uncertain at the time of sampling. For such uncertain taxonomy situations, see guidelines in Module 6 or the relevant section of the [OBIS Manual](#).

- **scientificNameID** (strongly recommended)

If you cannot obtain the required Life Science Identifier (LSID) from taxon matching with WoRMS then you must contact World Register Marine Species (info@marinespecies.org) to have an LSID created for your taxon. You will need to confirm that the species is marine. OBIS does not currently parse LSIDs from other sources (e.g., [Integrated Taxonomic Information System](#), [Catalog of Life](#)), but if you want to include other LSIDs alongside the WoRMS LSID, they must be specified in a predictable format.

- **occurrenceStatus**

Because occurrenceStatus is a binary field (“presence” or “absence”) this field can usually be easily inferred by data. If there are associated measurements or a record of an observation, the taxon in question would be present. If a particular species/taxa is present in one sample, but missing from another, then you could identify that species as absent from the second sample.

- **basisOfRecord**

basisOfRecord distinguishes what type of record is in your data. For records pertaining to a collected or stored specimen, you must choose one of the following terms: **PreservedSpecimen**, **FossilSpecimen**, **LivingSpecimen**. For records pertaining to an observation in the wild, you should use **HumanObservation** (e.g., observation in the wild) **MachineObservation** (e.g., photograph, DNA sequences). For records pertaining to DNA-derived data, you should use **MaterialSample**. For records pertaining to literature data, **basisOfRecord** should always reflect the evidence upon which the Occurrence record was based. For example, a researcher’s record based on photographs should specify **MachineObservation**, otherwise specifications should be **HumanObservation** (see [relevant GitHub discussion](#)).

In this lesson we reviewed some common challenges you might encounter during data formatting. We focused on standardizing dates to ISO 8601, converting coordinates to decimal degrees, and addressing missing required fields.

You have completed Module 2! Please complete Exercises 2-1, 2-2, 2-3 if you haven't already, as well as Quiz 2 before moving on to Module 3, where we will learn how to create the Event, Occurrence, and extendedMeasurementOrFact tables.

Module 3: Formatting data tables

Site: [OceanTeacher Global Academy](#)

Course: Contributing and publishing datasets to OBIS (self-paced)

Book: Module 3: Formatting data tables

Table of contents

Module 3

Lesson 1: Mapping DwC terms

- Darwin Core mapping examples
- Including data as verbatim
- Lesson Summary

Lesson 2: How to format Event table

- Steps 1-4
- Steps 5-8

Lesson 3. How to format Occurrence table

- Steps 1-5

Lesson 4: How to format extendedMeasurementOrFact table

- Steps 1-5
- Steps 6-8

End of Module

Introduction

In this module, you will learn how to format the core and extension data tables so that they adhere to Darwin Core standards. This will include how to map data field names to Darwin Core terms.

Learning Outcomes

After successful completion of this module, you should be able to:

- Map (or rename) original data field names to Darwin Core terms
- Identify which data goes into Occurrence, Event, and extendedMeasurementOrFact tables
- Restructure original data to follow DwC standards by creating Core and extension tables

How to Proceed

To succeed in this Module, you need to successfully complete the following lessons and exercises:

- Lesson 1: Mapping DwC terms
- Lesson 2: How to format Event table
- Lesson 3: How to format Occurrence table
- Lesson 4: How to format extendedMeasurementOrFact table
- [Exercise 3-1: Create the Event core and Occurrence extension](#)
- [Exercise 3-2: Create the eMoF extension table](#)

as well successfully complete Quiz 3 with a score of $\geq 80\%$

- [Quiz 3](#)

Mapping terms to Darwin Core

Throughout this course thus far we have learned about Darwin Core terms and the importance of using standardized terminology. There are many possible ways to set up a datasheet, and if you are new to OBIS you likely did not use controlled Darwin Core (DwC) terms or vocabulary before samples were collected. Thus you will likely have to map your data fields to DwC. We recommend documenting your choices in a Documentation sheet so you have a reference to go back to should the need arise. Such a document should include notes on the choices you made, as well as any actions you had to take (e.g. separate one column into many, convert dates or coordinates, etc.). You should also **always save one copy of the original dataset** before any changes were made.

Mapping as many of your data fields to DwC before creating the core and extension tables will help make this process easier, so we will look at some examples of commonly used data field names and the recommended DwC term(s) you can map to.

The following table provides non-exhaustive examples of original data field names and their associated DwC term. For a complete list of potential DwC terms, see the [Darwin Core Quick Reference Guide](#). If you ever have difficulty in mapping your data fields to a Darwin Core term, please do not hesitate to reach out to the OBIS helpdesk@obis.org or ask questions on the OBIS Slack (link in Course Resources).

Original term(s)	DwC term(s)
Date, time	eventDate
Species, g_s, taxa	scientificName
Any biotic/abiotic measurements	measurementType, measurementValue, measurementUnit (reviewed later in this module)
Depth	maximumDepthInMeters and/or minimumDepthInMeters
Lat/Latitude, Lon/Long/Longitude, dd, coordinates	decimalLatitude, decimalLongitude
Location	locality
Presence, absence	occurrenceStatus
Type of record/specimen	basisofRecord
Person/ people that recorded the original Occurrence	recordedBy
ORCID of person/ people that recorded the original Occurrence	recordedByID
Person/ people that identified the organism	identifiedBy
ORCID of person/ people that identified the organism	identifiedByID
Data collector, data creator	recordedBy
Taxonomist, identifier	identifiedBy
Record number, sample number, observation number	occurrenceID

Very often it is helpful to keep data as it was originally documented in "verbatim" DwC fields. This ensures anyone looking at your dataset will understand how the original data was interpreted. For example, if coordinates were originally recorded in degrees, minutes, seconds, it is good practice to keep this original field and rename it to `verbatimLatitude/verbatimLongitude/verbatimCoordinates`, as applicable.

There are a number of other verbatim fields you may consider including:

- [verbatimEventDate](#)
- [verbatimLocality](#)
- [verbatimDepth](#)
- [verbatimElevation](#)
- [verbatimIdentification](#)
- [verbatimLabel](#)

Note that you do not have to map every field in your original data to DwC for publishing. For example, a column for notes taken in the field or locations of specimens within storage facilities might not be relevant for publishing, and can be excluded from this mapping process.

In this lesson we have learned how to map (or rename) original data fields to DwC terms.

Now that we know how to create identifiers, how to standardize data, and how to map our fields to DwC, let's learn how to format a data table, starting with the Event core.

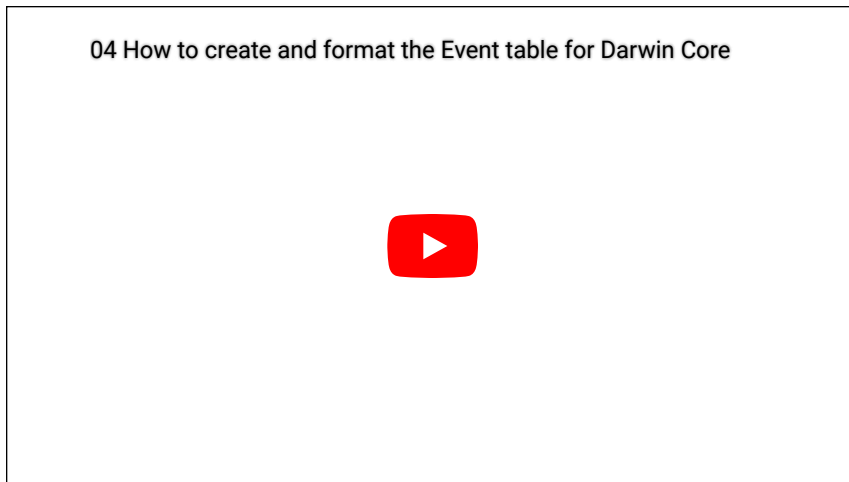
Formatting Event Tables

If your dataset uses an Event core structure (see Module 2), follow this process to format the data table:

1. **Create eventIDs** for each record.
2. **Add and fill** the **parentEventID** and **eventRemarks** fields as applicable
3. Identify the hierarchical event structure in your data, if present and **create** new records for parent events
 - o Add and fill any other relevant fields
4. **Identify** all columns in your data that will match with Darwin Core Event fields
 - o Include any relevant abiotic measurements (ENV-DATA) related to sampling events (e.g. sampling protocols). We will add these to the eMoF table later.
5. **Copy** these columns to a new sheet and name it Event
6. **Delete duplicate data** so only unique events are left
7. Ensure dates, times, and location information is **standardized** (as discussed in Module 2)
8. **Map** any remaining fields to Darwin Core

We will refer to using Microsoft Excel to reorganize data tables, but you can use other spreadsheet softwares or use R/Python. It helps to format table cells to “text only” in Excel to avoid unwanted reconfiguration and changes in the original data by the software.

Let's watch a video demonstrating the steps above before taking a look at each in detail.



Play Video

(Available at https://youtu.be/jyy6QO_p7v8)

After watching the video, let's review each of the steps before moving on to the next lesson where we will learn how to format Occurrence tables.

Step 1: Create eventIDs

Your original dataset likely contains many fields, some may be related to Event data and some will be related to Occurrence records. Before moving any fields or adding information, we must first ensure unique **eventIDs** are attached to each record. Create a copy of your original dataset (note: you should **never** make changes to your original data sheet! Keep it intact for later reference), and add a new, blank column named " **eventID**". Follow guidelines from Module 2 to **create unique eventIDs** for records belonging to different sampling events if you do not already have an identifier for them. If you already have unique identifiers for each record, copy these identifiers into the eventID column.

As a reminder, all records belonging to the same sampling event will have the same **eventID**. It is okay to have records with the same **eventID** at this step.

Step 2-3: Identify event hierarchy, create parentEventID and eventRemarks fields

Now that unique identifiers have been created, **identify the hierarchy levels** in your events. What types of events are nested in each other? Which event type would be at the highest level, the lowest? You might not yet have records for events higher in the hierarchy (i.e. parent events). This is okay, we will create them. It is easier if you recall the rationale from your sampling design to build the events and parent events. If you were not the one that developed the study, you may find this information in related documentation, such as in the manuscript, project report etc.

Add two new columns named parentEventID and eventType. For each of your records, indicate what type of event occurred - a subsample, sample, a site or station visit, a cruise, a deployment, etc. Write this event type in the **eventType** field and try to use suggested controlled vocabulary. You may also include event information in **eventRemarks**.

Now that you have identified your event hierarchy structure, we need to create records for the events at the higher hierarchy levels, the parent events. Add new records for each parent event as applicable. The other data fields (**eventDate**, **decimalLongitude**, etc.) can either be left blank or filled in as relevant. You should provide the most relevant information for each sampling event. Information provided to parent events will be passed to child events only when the child event fields are blank.

For example, coordinates do not need to be repeated in a parent event if the samples have more specific coordinates, but you are welcome to also provide generalized coordinates for parent events if available. However, if many samples were taken at one location, then a parent event (e.g. station) may contain the coordinates for all child events, thus leaving the sample event records with blank coordinate fields. However do note that it is okay to populate parent *and* child events with the same information particularly if the dataset will be harvested by GBIF, because GBIF does not currently pass parent event information to child events.

Step 4: Identify relevant Event fields

Now that all records have attached **eventIDs** & **parentEventIDs**, let's ensure only our Event table only contains event-specific data. As a reminder, an Event table is required to include the following terms:

- **eventDate**
- **eventID**
- **parentEventID** (if applicable)
- **decimalLatitude**
- **decimalLongitude**

Other terms you should consider adding are grouped by their associated Darwin Core class below:

- Class Event | DwC:eventRemarks
- Class Event | DwC:eventType
- Class Event | DwC:year
- Class Event | DwC:month
- Class Event | DwC:day
- Class Event | DwC:type
- Class Location | DwC:country
- Class Location | DwC:island
- Class Location | DwC:coordinateUncertaintyInMeters
- Class Location | DwC:countryCode
- Class Location | DwC:footprintWKT
- Class Location | DwC:geodeticDatum
- Class Location | DwC:islandGroup
- Class Location | DwC:locality
- Class Location | DwC:locationAccordingTo
- Class Location | DwC:locationID
- Class Location | DwC:locationRemarks

- Class Location | DwC:maximumDepthInMeters
- Class Location | DwC:minimumDepthInMeters
- Class Location | DwC:stateProvince
- Class Location | DwC:verbatimCoordinates
- Class Location | DwC:verbatimDepth
- Class Location | DwC:waterBody

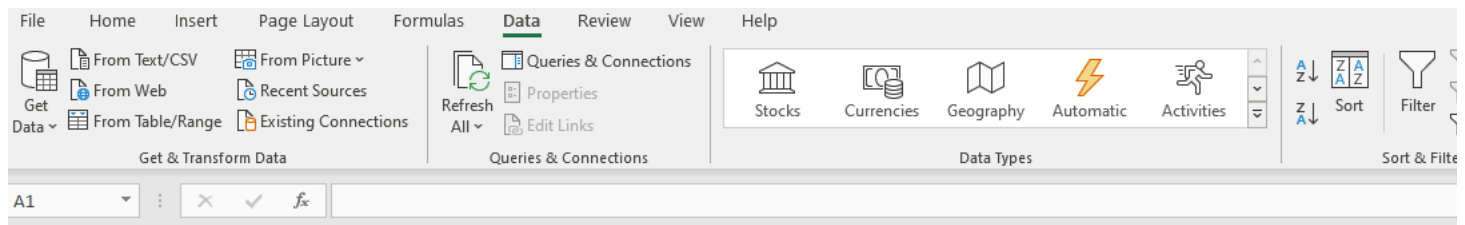
Look for and identify which of your data columns match or relate to any of these terms and then move to step 5. These terms (or columns) are not required to upload your data into OBIS, however, consider including them if you have the information.

Step 5: Copy relevant fields

Create a new, blank spreadsheet and rename it to "Event". You will then have 2-3 sheets in your spreadsheet file: the original data untouched (or saved in a separate file altogether, the working copy of the original datasheet that we just added *eventIDs* to, and a blank sheet called Event. Next, select all the previously identified relevant Event fields from the working sheet, and copy them into the Event datasheet. For now, please **include any fields related to any measurements regarding your events** (biotic measurements on observations will be handled in the next lesson on Occurrence data tables) - this may include sampling protocols, sampling gear, temperature, salinity, etc. Be sure that the measurement is associated with the *event*, and not an *occurrence* (e.g. some temperature data or other abiotic measurements might be associated with an occurrence rather than an event). Measurement (meta)data will be organized separately after the Event and Occurrence tables are completed, but it will be easier to associate them with the proper *eventID* by doing this step.

Step 6: Delete duplicate data

Use Microsoft Excel's **Remove Duplicates** function from the Data tab on the ribbon to ensure there are no repeated *eventIDs* in this table and only unique events are left.



Step 7: Fill in other relevant fields and standardize information

Fill in any other event fields with relevant information. Then confirm that dates, location information, etc. all match with data standards (ISO 8601 for dates) as we learned in Module 2.

Step 8: Map unmapped terms to Darwin Core

You must rename your columns to match with Darwin Core terms. As mentioned previously, we recommend keeping notes on this process by creating a separate sheet and documenting the original field name, the Darwin Core name you mapped it to, and your reasons why. Call this the "**Documentation sheet**" and make use of it as you proceed with the data formatting process. It is very important to list and describe all the modifications carried out along with formatting and curating your data tables. It will help you track not only the quality of data but also date back if any procedure was mistaken.

Keep in mind you may need to split one column into two or more to match with Darwin Core terms, or combine multiple columns into one. See below for an example of the Documentation sheet for Event data.

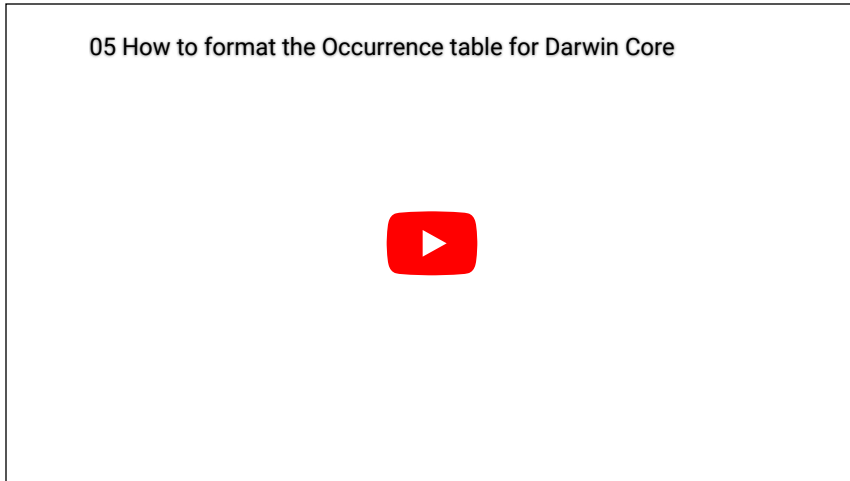
	A	B	C
1	Original Field Name	DwC Name	Notes
2	date, time	eventDate	Combined date and time columns into eventDate
3	lat	decimalLatitude	Converted from ddmms to decimal degrees
4	lon	decimalLongitude	Converted from ddmms to decimal degrees
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			

We will practice formatting an Event table in Exercise 3-1. But first let's learn about the Occurrence table in the next lesson.

If your dataset structure will have an Occurrence core, or has an Occurrence extension (all OBIS data have at least one occurrence record associated, regardless of what organization structure you have chosen), you should follow the below procedure to format your file.

1. **Identify** columns in your raw data that match with Occurrence fields
 - Include columns with measurements for now, but they will be moved to an eMoF table(s)
2. **Copy** these columns to a new sheet named Occurrence (it is good practice to never make changes to your original datasheet)
3. **Create and add** `occurrenceIDs` for each unique occurrence record
4. **Add and fill** `basisOfRecord` and `occurrenceStatus` fields
5. Ensure your column names **map to Darwin Core terms**

Let's take a closer look at each of these steps by first watching the video below.



Play Video

(Available at https://youtu.be/G_AmAmS7ILc)

Now let's look at each step in detail.

Step 1: Identify relevant fields

Begin by **identifying all fields relevant to observations or occurrences** in the working copy of the datasheet. These may include taxon names, biotic measurements, identification information, hosting institution information, etc. You will need to map your raw data terms to Darwin Core terms if they haven't been yet, but first, let's look at which terms are most relevant for an Occurrence core or extension table. There are several Darwin Core terms that are **required** in Occurrence data tables, including:

- [occurrenceID](#)
- [eventID](#)_(required for Occurrence extension, not required for Occurrence Core)
- [occurrenceStatus](#)
- [basisOfRecord](#)
- [scientificName](#)
- [scientificNameID](#)_(strongly recommended)
- [eventDate](#)_(not required for Occurrence extension, required for Occurrence Core)
- [decimalLatitude](#)_(not required for Occurrence extension)
- [decimalLongitude](#)_(not required for Occurrence extension)

While these are the bare minimum, you should strongly consider adding other terms if you have the corresponding information/data in your dataset or documentation. These are identified by their associated Darwin Core class below, and found [here](#).

- Class Occurrence | DwC: [associatedMedia](#)
- Class Occurrence | DwC: [associatedReferences](#)
- Class Occurrence | DwC: [associatedSequences](#)
- Class Occurrence | DwC: [associatedTaxa](#)
- Class Occurrence | DwC: [preparations](#)
- Class Occurrence | DwC: [recordedBy](#)
- Class Occurrence | DwC: [materialSample](#)
- Class Occurrence | DwC: [materialSampleID](#)
- Class Record | DwC: [bibliographicCitation](#)
- Class Record | DwC: [catalogNumber](#)
- Class Record | DwC: [collectionCode](#)
- Class Record | DwC: [collectionID](#)
- Class Record | DwC: [dataGeneralizations](#)
- Class Record | DwC: [datasetName](#)
- Class Record | DwC: [institutionCode](#)
- Class Record | DwC: [modified](#)
- Class Taxon | DwC: [kingdom](#)
- Class Taxon | DwC: [scientificNameAuthorship](#)
- Class Taxon | DwC: [taxonRank](#)
- Class Taxon | DwC: [taxonRemarks](#)

Any fields you have related to biotic measurements (e.g., sex, lifestage, biomass, length) will also be included in the extendedMeasurementOrFact table. Such measurements can remain in the Occurrence table as long as they can be mapped to the appropriate DwC term (e.g. DwC:Occurrence:sex). This is important because not every data aggregator outside of OBIS indexes the eMoF table, so otherwise this information may be lost.

Step 2: Copy relevant fields

Create a new, blank spreadsheet and rename it to "Occurrence". After identifying the relevant data columns in Step 1, copy them into the new sheet. Make sure to include the **eventIDs** so occurrence records can be linked back to the Event table. We would like to again emphasize that you should **never** make changes to your original data sheet. It is important to keep the original data intact in case you need to go back to reference data. This is why we are using a working copy of the data.

Step 3: Create occurrenceIDs

Add a new, blank column and name it "occurrenceID". Follow guidelines from Module 2 to create unique **occurrenceIDs** for each record if you do not already have an identifier for them. If you already have unique identifiers for each record, copy these identifiers into the **occurrenceID** column. Remember **occurrenceIDs** can be created by adding information to an **eventID**.

Step 4: Fill basisOfRecord and occurrenceStatus

Create two new columns for **basisOfRecord and **occurrenceStatus**** and populate these for each record. As a reminder, **occurrenceStatus** must be either **present** or **absent** and **basisOfRecord** must be one of: **PreservedSpecimen, FossilSpecimen, LivingSpecimen, MaterialSample, Event, HumanObservation, MachineObservation, Taxon, Occurrence, MaterialCita**

tion.

Step 5: Map terms to Darwin Core

As mentioned previously, column headers must map to Darwin Core terms if you have not already done so. Continuing to build on the Documentation sheet, you might add comments such as below.

	A	B	C	D
1	Original Field Name	DwC Name	Notes	
2	date,time	eventDate	Combined date and times columns into eventDate	
3	lat	decimalLongitude	Converted from ddmms to decimal degrees	
4	lon	decimalLongitude	Converted from ddmms to decimal degrees	
5	species	scientificName	fixed minor typo in species name and recorded in "scientificName"	
6	species	verbatimIdentification	Renamed "species" to verbatimIdentification	
7		occurrenceID	created IDs by combined eventID with unique identifiers	
8				
9				
10				

workingsheet | Documentation | Occurrence | Event | (+) |

Ready | 97% Accessibility: Good to go

Practice creating the Event and Occurrence table in Exercise 3-1. Once you're done, return to Lesson 4 to learn how to format the extendedMeasurementOrFact table.

Formatting the eMoF

To review, any data related to abiotic or biotic measurements, including sampling information and protocols should be included in the extendedMeasurementOrFact (eMoF) table. Measurement data could also go into the [MeasurementOrFact](#) (MoF) extension, however OBIS recommends using the [extendedMeasurementOrFact](#) instead, particularly if your data is based on an Event core table.

Structuring your data to fit the eMoF extension may be one of the more confusing processes during data formatting. Rather than documenting each of your measurements in separate columns (e.g., columns for biomass, abundance, length, gear size, percent cover, etc.), these measurements will be condensed into one column: `measurementValue`.

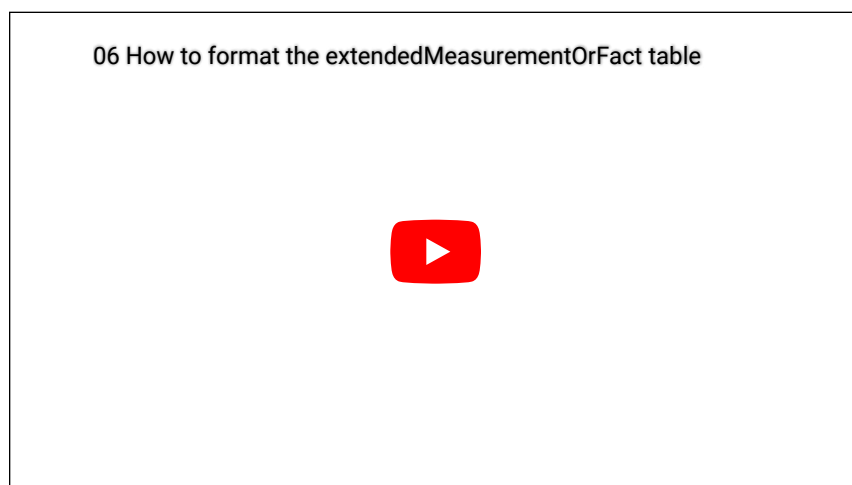
Then `measurementType` describes what the measurement actually is, for example whether it is an abundance value, length, percent cover, or any other biotic/abiotic measurement. `measurementUnit` is used to indicate the unit of the measurement.

By linking `measurementType` and `measurementValue` with the identifiers `eventID` and/or `occurrenceID`, you can have measurements linked to *one* event (e.g. temperature, salinity), measurements linked to occurrence records (e.g. length, weight), as well as sampling facts that are linked to events (protocol, size, gear, etc.). Information specifically related to how samples were taken will have the `measurementTypes`: `sampleSizeValue`, `sampleSizeUnit`, `samplingEffort`, and `samplingProtocol`.

Similar to Event and Occurrence tables, we can follow this process to build our eMoF extension table:

1. **Create** a blank sheet and name it eMoF
2. **Add** 9 column headers for:
 - `eventID`, `occurrenceID`, `measurementType`, `measurementValue`, `measurementUnit`, `measurementTypeID`, `measurementValueID`, `measurementUnitID`, `measurementRemarks`
3. **Copy** `eventID` and `occurrenceID` values for measurements from your [Occurrence](#) table and paste into the associated fields in your new, blank eMoF table
 - Note that measurements associated with an occurrence **must** have the corresponding `occurrenceID` and `eventID`, measurements associated with an event must have the corresponding `eventID`
4. Copy the first column containing **measurement values**, paste into the `measurementValue` field
 - Then, fill `measurementType` with the name of the variable (e.g., count, length, etc.)
5. **Add the unit** of measurements, where applicable, to the `measurementUnit` field (e.g. individuals, cm, lbsetc.)
6. For any other measurements related to the occurrences (or events), **repeat** steps 3-5
7. **Repeat** steps 3-6 for any measurements in the **Event** table
8. **Fill measurement ID fields** `measurementTypeID`, `measurementUnitID`, and `measurementValueID` (details on how to find these identifiers will be covered in Module 5)

Let's watch a video demonstration of this process.



Play Video

(Available at <https://youtu.be/EjM0HRrF1B4?si=nh5RRgU3KuDUrbi>)

Now we will review again each step in this process.

Step 1-2: Create a blank sheet and add 9 column headers

In your spreadsheet software, **create a blank sheet named "eMoF"**. Label 9 columns with the headers: `eventID`, `occurrenceID`, `measurementType`, `measurementValue`, `measurementUnit`, `measurementTypeID`, `measurementValueID`, `measurementUnitID`, `measurementRemarks`

Step 3: Copy identifiers from the Occurrence table

Go to your Occurrence table and **copy all eventIDs and occurrenceIDs** (i.e. copy the entire column). **Paste** these into the `eventID` and `occurrenceID` column of the eMoF table. This ensures the measurements we will paste will be associated with the correct event and occurrence record.

Step 4-5: Copy the first measurement variable and add units

From your Occurrence table, identify the first column that contains measurement data. It doesn't matter which one you choose first, but it's important to do one measurement at a time. **Copy (or cut) the column of measurement values and paste into the measurementValue** column in the eMoF sheet. In the `measurementType` column, be sure to indicate for each row what the type of measurement was. You can use the original column header to fill this information in (e.g. length).

Then in the `measurementUnit` column, fill in the the units for this measurement variable, being sure to include every record.

Step 6: Repeat steps 3-5 for additional measurements

We will repeat the previous steps for any additional columns containing measurements on your occurrences. Begin with step 3 and copy the `eventID` and `occurrenceID`. **Paste these IDs below the last record.** This will stack the measurements into a "long" format.

Then, **copy or cut the next set of measurement values**, pasting again into `measurementValue`. Fill in the `measurementType` and `measurementUnit` columns as before. **Continue to repeat this process** for all measurements until all have been moved into the eMoF table.

Step 7: Repeat for Event measurements

Repeat steps 3-5 for any measurements in your Event table, ensuring to copy the corresponding `eventIDs`. Recall that measurements associated with an event **must** have the corresponding `eventID`, but they may not need a corresponding `occurrenceID`.

Step 8: Fill measurement ID fields `measurementTypeID`, `measurementUnitID`, and `measurementValueID`

After you have completed formatting your measurement data, it is important to add identifiers, or Unique Resource Identifiers (URIs), associated with the measurement type, values, and units. This will help account for the heterogeneity that occurs when filling the fields, particularly the `measurementType` field. For example, there are many ways one may write "length" (len, length, fork length, Length, etc.). By attaching the same URI or identifier to your measurements, we know that all measurements with the same identifier are the same type of measurement.

For now, all you need to know is that you should try to find a `measurementTypeID` URI that belongs to NERC's [P01 collection](#). We will come back to specific guidelines for choosing measurement ID identifiers in Module 5.

In this Module we learned how to format the Event, Occurrence, and eMoF tables. We learned about identifying hierarchical event structure and removing duplicate event records, how to identify occurrence data within a dataset, and that formatting the eMoF likely entails switching data from a wide or unstacked format (where each measurement was a separate column) to a long or stacked format (measurements are stored in the `measurementType`, `measurementUnit`, and `measurementValue` columns).

Practice what you have learned by completing Exercise 3-1 if you haven't already done so, and Exercise 3-2 to create an eMoF table. Complete Quiz 3, then, in the next module we will learn how to format a special type of data to Darwin Core: DNA derived data.

Module 4: DNA derived data

Site: [OceanTeacher Global Academy](#)
Course: Contributing and publishing datasets to OBIS (self-paced)
Book: Module 4: DNA derived data

Table of contents

Module 4

Lesson 1: Introducing DNA data

- Understanding DNA-based occurrence
- Raw DNA sequences
- MIXS and DwC
- Lesson summary

Lesson 2: Types of genetic data

- Category I: DNA-based occurrences
- Category II: Enriched occurrences
- Category III: Targeted species detection
- Category IV: Name references
- Category V: Metadata-only
- Lesson Summary

Lesson 3: Compiling DNA data for categories I and II: eDNA and metabarcoding

- DNA data files
- DNA files to Darwin Core
- Beginning data formatting
- Format Occurrence core table
- Format DNA Derived Data extension table
- Format eMoF table
- Taxonomic sequence names: known or unknown sequences
- Lesson summary

Lesson 4: Compiling DNA data for category III: qPCR data

- Format Occurrence core for qPCR data
- Format DNA derived data extension for qPCR data
- Lesson summary

Lesson 5: Categories I and II example: eDNA

- Format eDNA example: Occurrence core
- Format eDNA example: DNADerivedData extension
- Where to find information
- Lesson summary

End of Module



Introduction

In this module you will learn how to format DNA derived data. This includes data obtained from eDNA sampling, qPCR, or DNA from individuals. We will follow an example to help demonstrate the formatting process, and highlight areas you should be particularly careful of.



Learning Outcomes

After successful completion of this module, you should be able to:

- Understand the different categories of genetic data
- Prepare DNA and sequence data for publishing to OBIS by
 - Using an Occurrence core
 - Recording genetic information in the DNA-Derived Data extension
- Identify relevant fields for eDNA/metabarcoding data compared to qPCR data



How to Proceed

To succeed in this Module, you need to successfully complete the following lessons and exercises:

- Lesson 1: Introducing DNA data
- Lesson 2: Types of genetic data
- Lesson 3: Compiling DNA data for categories I and II: eDNA and metabarcoding
- Lesson 4: Compiling DNA data for category III: qPCR data
- Lesson 5: Categories I and II example: eDNA
- [Exercise 4-1: Describe DNA data fields](#)

as well successfully complete Quiz 4 with a score of $\geq 80\%$.

- [Quiz 4](#)

Introduction to DNA data

DNA derived data are increasingly being used to document taxon occurrences. This genetic data may come from a sampling event, an individual organism, may be linked to physical material (or not), or may result from DNA detection methods e.g., metabarcoding or qPCR (Quantitative Polymerase Chain Reaction). Thus genetic data may reflect a single organism, or may include information from bulk samples with many individuals. Still, DNA-derived occurrence data of species should be documented as standardized and as reproducible as possible.

To ensure DNA data are useful to the broadest possible community, a community guide entitled [Publishing DNA-derived data through biodiversity data platforms](#) was published by GBIF, OBIS, and others. This guide uses the [DNA derived data extension for Darwin Core](#), which incorporates MIXS (Minimum Information about any (X) Sequence) terms into the Darwin Core standard. We will review important aspects of this guide and extension in this module.

Note that because guidelines for DNA data are relatively new, there may be updates to the publishing guide in the future.

Occurrences from DNA sampling are by nature derived data, most commonly *not* resulting from a direct detection either by visual methods or by catching individuals. Therefore, many decisions are made in the process of collecting and analyzing DNA information that may affect the results. To ensure that in the future we can evaluate if the DNA data is reliable and comparable to other studies, it is important to record as much metadata on these sampling, sample processing, and data analysis choices as possible. The DNA-derived data extension to Darwin Core was developed to allow this information to be recorded.

It is important to understand what the recorded species information in the DNA derived data extension represents. The amount of information that can be acquired from DNA sampling and next generation sequencing methods is huge, often leading to lists of hundreds of species and thousands of sequences per sample. Sequences (or “reads”) acquired from the sequencing machine, that have not been analyzed, are called “raw” sequences, and still require in-silico analysis by bioinformatics before they can be given species names.

Let's understand more about these raw DNA sequences, but first see the figure below for a simplification of the process from DNA sample collection to species identification.

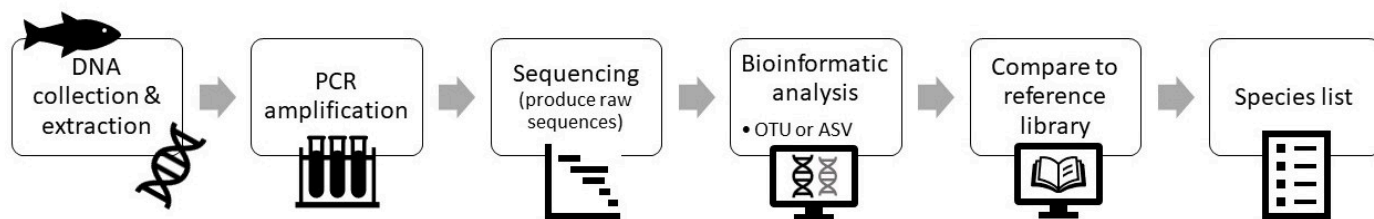


Figure adapted from Figure 2 in *Publishing DNA-derived data guide* (Abarenkov et al. 2023 <https://doi.org/10.35035/doc-vf1a-nr22>).

Raw sequence information is already commonly shared publicly to sequence databases in the International Nucleotide Sequence Database Collaboration (INSDC). There are three databases in the collaboration; the National Center for Biotechnology Information in the USA (NCBI), the European Bioinformatics Institute of the European Molecular Biology Laboratory (EMBL-EBI), or the DNA Data Bank of Japan (DDBJ). Sequences in these databases can be openly accessed, analyzed, and used for comparisons. However, as they are not processed, it is not usually possible to search for species, location, sampling time, etc., because the data still requires a complex analysis.

To acquire species information the raw sequences must be analyzed. This commonly includes quality control, error correction, and taxonomic annotation. Taxonomic annotation is done by comparing the sequence to a reference database containing known named sequences from the same genetic region. Finally, the quality controlled sequences (i.e. true sequences) are called **Amplicon Sequence Variants (ASVs)**. If the sequences are then *also* clustered to form larger groups (i.e. using specific algorithms or percent similarity) they are called **Operational Taxonomic Units (OTUs)** and can be interpreted to represent a taxonomic group like a species.

ASVs contain more information as they have not been clustered, even if the sequences differ by one nucleotide. On the other hand, OTUs are usually clustered based on 97% sequence similarity.

What the DNA-derived data extension allows for, is recording and storing the ASV and OTU information linked to species annotations and including rich metadata. This ensures that the data can be searched and accessed in a more flexible manner, and therefore opens up a host of different possibilities for large data analysis.

As we mentioned earlier, MIXS standards were incorporated into the Darwin Core guidelines. Let's review these terms now.

An important note is that data to INSDC and EMBL-EBI is deposited using the Minimum information about any (x) nucleotide sequence (MiXs, Yilmaz et al, 2011), the core standard developed by the Genomics Standard Consortium (GSC). This standard consists of checklists for describing genomes (MIGS), metagenomes (MIMS) and marker sequences (MIMARKS). As mentioned, a [mapping between DwC and MiXs](#) has been made to ensure interoperability of the two standards, see the table below for how the terms compare.

MiXs terms	Associated DwC terms
lat_lon	verbatimCoordinates decimalLatitude decimalLongitude verbatimLatitude verbatimLongitude
depth	verbatimDepth minimumDepthInMeters maximumDepthInMeters
alt	minimumDistanceAboveSurfaceInMeters maximumDistanceAboveSurfaceInMeters
elev	verbatimElevation minimumElevationInMeters maximumElevationInMeters
geo_loc_name	country waterBody higherGeography
collection_date	eventDate
source_mat_id	materialSampleID
biotic_relationship	associatedOrganisms
samp_collect_device	samplingProtocol
samp_mat_process	samplingProtocol
samp_size	sampleSizeValue sampleSizeUnit

url	references associatedMedia associatedOccurrences associatedOrganisms associatedReferences associatedSequences associatedTaxa
-----	--

In this lesson we began learning about DNA-derived data, what it is, what are raw sequences, and we gained some basic understanding that DNA data needs to be processed before being formatted to DwC.

Next, let's learn about the five types of DNA-derived data, and which ones can be published to OBIS.

Types of DNA data

Before we get into details about how to format DNA-derived data, it is good to recognize that there are 5 categories for which genetic data could fall into:

1. DNA-derived occurrences
2. Enriched occurrences
3. Targeted species detection
4. Name references
5. Metadata only

The figure below provides a graphical representation of each of these categories.

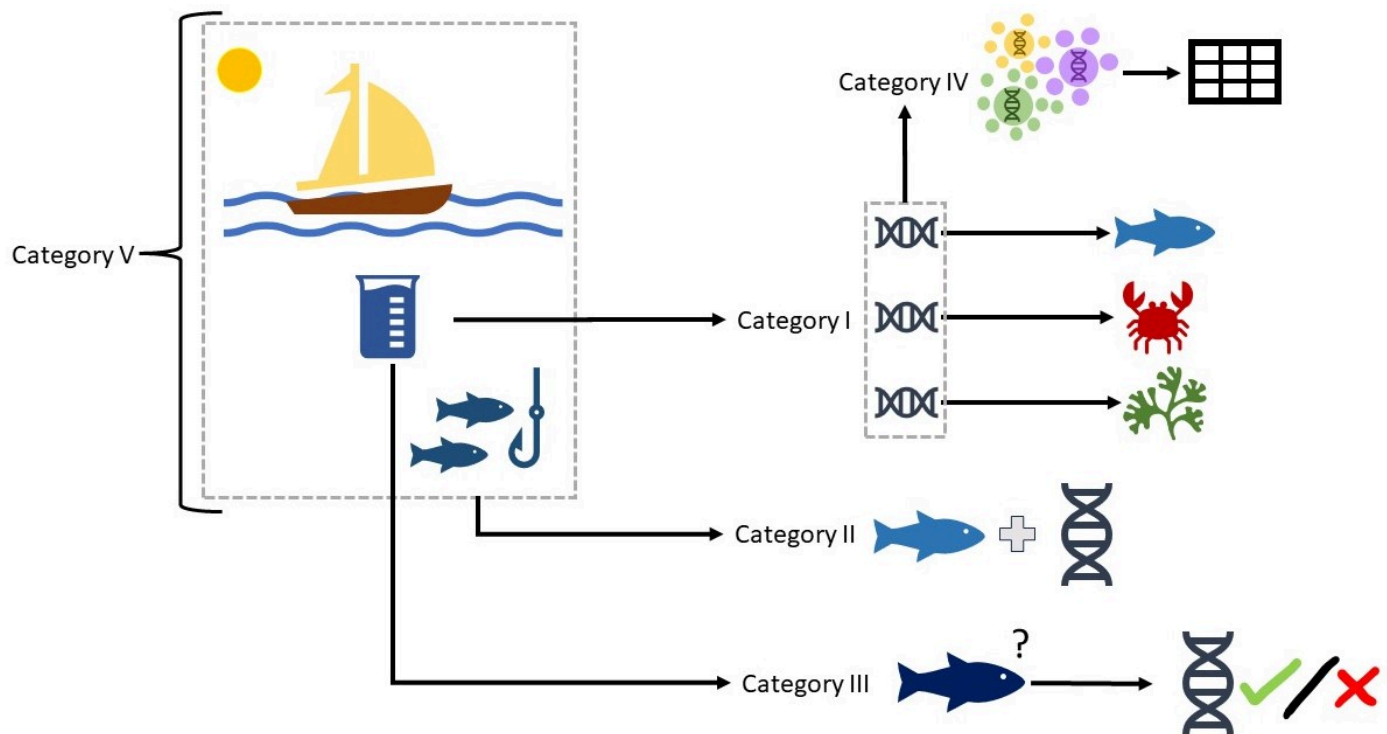
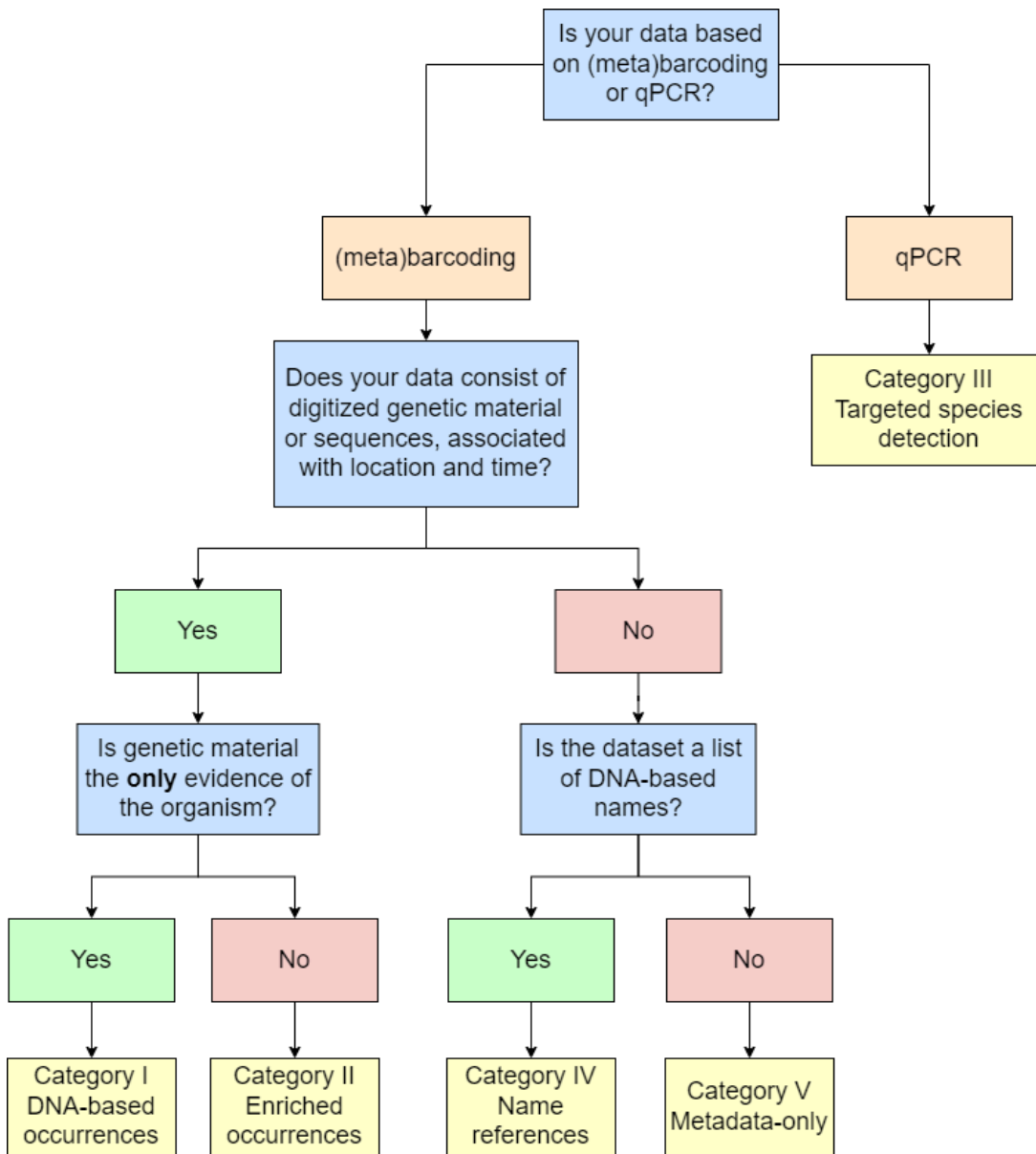


Figure adapted from Figure 6 in *Publishing DNA-derived data guide* (Abarenkov et al. 2023 <https://doi.org/10.35035/doc-vf1a-nr22>).

But how do you know which type of data you have?

To determine which category your data falls into, follow the decision tree below, (adapted from the [Data packaging and mapping](#) section of the *Publishing DNA-derived data guide*).



We will review formatting guidelines for eDNA (Category I) and qPCR (Category III) data in Lessons 3 and 4, respectively. First, let's take a closer look at the five categories, and what kind of data falls into each.



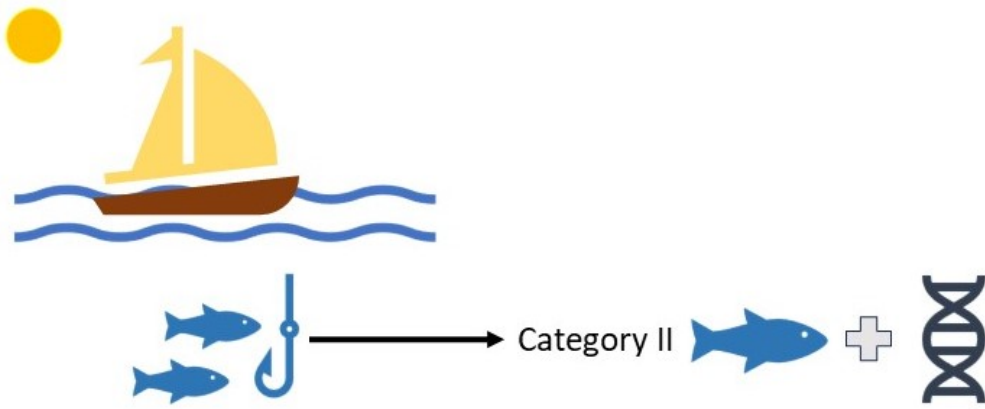
This category contains data where **the only evidence of an occurrence or presence of an organism is through the detection of its DNA** through sequencing. In these cases, no observation of a specimen occurred - the detection is solely DNA-based.

Commonly, category I includes DNA that is sampled directly from the environment (e.g. water, soil, and air samples), but can also be, for example, an analysis of stomach contents of an organism. However, the occurrences in this category have not been detected directly and therefore it is highly important to understand how the species information was acquired, and to evaluate the reliability and comparability of the data.

In principle, ASVs/OTUs can be the only indication of species presence - a species annotation is not required. Currently, reference databases are still incomplete (i.e. missing sequences for many species) and therefore many sequences will remain unknown. The most important information to record is each individual sequence. These sequences can then be easily compared and analysed in future studies.

It is also important to note, that due to the nature of sequencing, all quantities of reads recorded in this category are relative; i.e. the quantity is always related to the total amount of reads acquired in each sample, and does not directly reflect the quantity of that sequence in the original sample. So for each occurrence it is required to record the amount of that specific sequence that was detected in a sample, as well as the total amount of reads in that sample (e.g. indicating that 12% of all reads had this sequence).

Metagenomics, metabarcoding, and eDNA works typically fall into this category.



An “enriched occurrence” is when **genetic material is associated with an observation or a specimen**. A DNA sequence is not the only evidence of the occurrence, rather the associated specimen was also observed in some way (e.g., physical observation, collected specimen, etc.).

Studies that contain reference material and genetic sequences fall into this category, for example barcoding of museum specimens and some metabarcoding studies.

Datasets belonging to Categories I and II are usually similar in data formatting and thus will have similar recommendations. We will look at an example of how to format such data with an eDNA use case in Lesson 5.



This category contains datasets that have used a targeted qPCR/ddPCR assay to **detect the presence (or absence) of a specific organism in a sample**. This approach is frequently used to detect the presence of invasive species (e.g. [American bullfrog](#), [Ficetola et al. 2008](#)) and thus can be a very valuable genetic tool. Observations detected through qPCR are (likely) not associated with the genetic material (i.e. the sequence). With qPCR an occurrence is recorded when a known sequence is found in the sample using PCR. The analysis does not include sequencing of the PCR products, therefore the analysis (and metadata) requirements differ considerably from category I and II. Importantly, the genetic material itself (e.g., a DNA sequence) is not usually associated with the occurrence record because the analysis does not include sequencing.

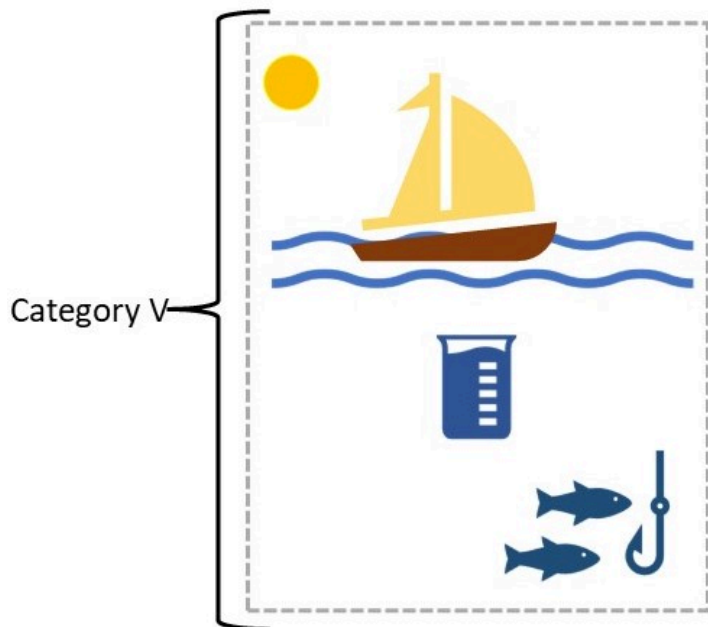
Data in this category are sensitive to the detection limit of the assay (the minimum amount of DNA required to detect presence of target DNA), as well as contamination from field and lab procedures. This can make it difficult to have high confidence in an absence. When preparing data in this category, it is important to provide enough information to be able to evaluate the records.



In reference databases, **sequences can be clustered by similarity and given an ID**. These IDs can then be used to index other unknown sequences, similar to how OBIS uses `scientificNameID` to categorize species names. These types of IDs can be registered in GBIF as Name References, but because they, by definition, do not have corresponding Linnaean names (e.g. a taxonomic identification), we do not currently publish this data to OBIS. This does not apply to unknown sequences in a typical study, but only the list of names defined in well-established stable genetic reference databases.

Datasets from this category are thus not relevant for OBIS at this time.

NOTE: if you have this type of data and want to publish it to GBIF, please see the [publishing guide for name references](#), otherwise we won't be focusing on this type of data in this course.



This last category contains **data about the data**, very similar to the metadata we will learn about in Module 7: Publishing to OBIS. It can include a description of the dataset, methods used in the field and/or laboratory, sequencing methods, authors and affiliations, taxonomic, temporal, and geographical scope, and the purpose of the dataset. In some cases, you may want to indicate that a study has been conducted and that the raw sequence data is available, but you might not have the occurrences (i.e. ASVs/OTUs) available yet. In these cases you can publish metadata only, and then add the occurrence information later when it becomes available. A link can be added to the raw sequences submitted to public sequencing databases with more rich metadata. Submitting Metadata-only datasets can be useful so that your data will be discoverable even before you have occurrence records, and can make the publishing process easier.

Guidelines for this type of data should follow general metadata recommendations in Module 7. Be sure to provide as many details as possible, especially when describing field, laboratory, and bioinformatics methods. Alternatively, provide a link to the protocols if they are already published in detail elsewhere (e.g. [protocols.io](https://www.protocols.io) or [NEON protocols collection](https://neonprotocols.org/)).

Now that we understand the different types of genetic data, let's look at the specific guidelines on how to format DNA-derived data so that it can be published to OBIS.

Regardless of what category your genetic data falls into (with the exception of Category V), datasets with genetically derived data must be published with Occurrence core, not Event core. Only one extension level can be added, and in this case it is the DNA derived data extension linked to each occurrence. A [new data model](#) is being developed by GBIF and the OBIS community that will provide a solution for this, however as it is not implemented yet, we will focus on the current Darwin Core recommendations. Each record (or row) in this Occurrence core table will contain information about the organisms (or unknown ASVs/OTUs) in the sample, including:

- The taxonomy and quantity of detected sequences
- The total quantity of the reads in a sample (to enable calculation of the relative abundance)
- Location where samples were collected
- Time when samples were collected
- References for the identification procedure
- The DNA sequence that defines that occurrence
- Links to the generated raw sequences

As indicated in Lesson 1, the [DNA-Derived Data extension](#) will be linked to the Occurrence core, by using `occurrenceID` and/or `eventID`. The DNADerivedData extension will contain important information about sequencing methods and the sequence (e.g., concentration, annealing temperatures, primer information, etc.). It is important to note that this extension may change as guidelines are refined. Thus using this extension may require remapping in the future.

In this lesson we will focus on how to compile and format data only for DNA data belonging to category I or II.

DNA data files

To format your genetically derived data to publish to OBIS, make sure you have the information on the sequence and possible taxonomy for each occurrence record associated with a DNA sample. It is good to know that genetic data is often recorded in multiple different files, and you might receive this type of format from your data provider, including: an OTU-table, a taxonomy table, a sample information table, and a .fasta file with sequences. The OTU-table is a sequence by sample table (Table 1), which records the quantity of each unique sequence found in each sample. Sequences are usually referred to by an ID, which is unique only in the dataset (e.g. asv1, asv2, asv3 ...).

Table 1. An example of an OTU/ASV table for ASVs (unique sequences are rows), indicating how many copies of each sequence was found in each sample (columns). In this case the column sums would give you the total read count for each sample. The table may also be transposed (OTUs/ASVs as columns and samples as rows).

	Sample1	Sample2	Sample3	Sample...
asv1	12560	123	0	...
asv2	4	3	0	...
asv3	459	45	650	...
asv...

The taxonomy table (Table 2) is a sequence by taxonomy table, which records the taxonomy linked to each unique sequence, as defined by the annotation method.

Table 2. The taxonomy table will typically record the taxonomy of each ASV/OTU as assigned in the bioinformatic analysis.

	Kingdom	Phylum	Class	Order	Family	Genus	Species
asv1	Eukaryota	Mollusca	Bivalvia	Ostreida	Ostreidae	Dendostrea	Dendostrea frons
asv2	Eukaryota	Chordata	Actinopteri	Chaetodontiformes	Leiognathidae	Equulites	
asv3	Eukaryota	Chordata					
asv...

The sample information table (Table 3) records the metadata of each sample (e.g. location, time, and collection method).

Table 3. An example of a typical sample information table, which will give you information to differentiate the samples in a dataset from each other.

	Date	Filter	Amount	Location	Depth	Replicate
Sample1	2023-07-05	0.2	1200	Site1	10	1
Sample2	2023-08-03	0.2	800	Site2	10	1
Sample3	2023-08-03	0.8	5300	Site1	100	1
Sample..

Finally the .fasta file records the actual DNA sequence that is linked to each sequence id. An example of a .fasta file format:

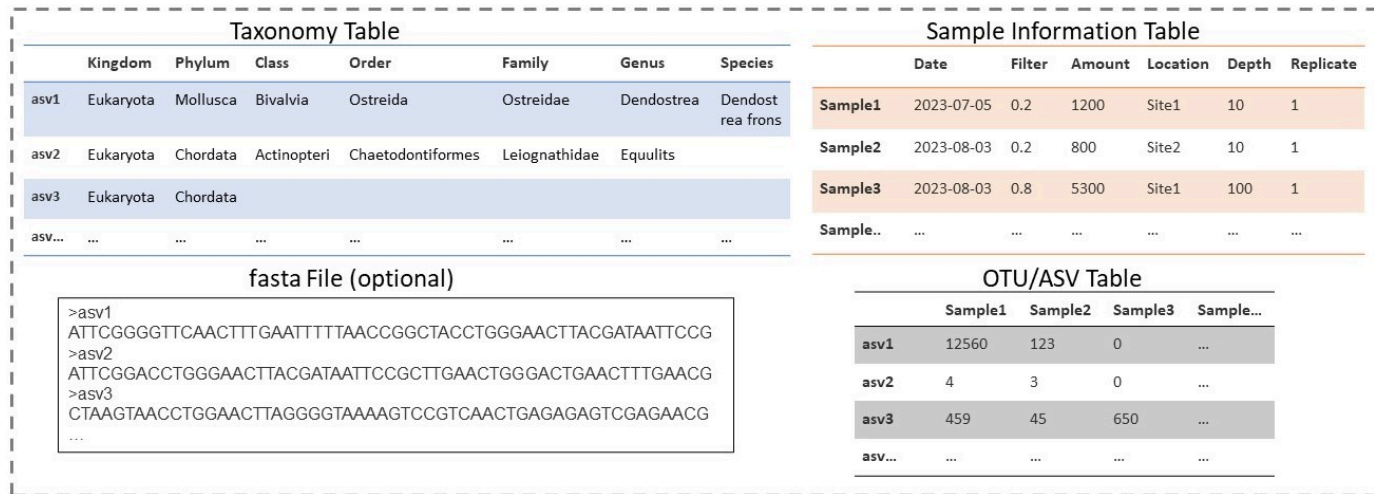
```
>asv1
ATTCGGGGTTCAACTTTGAATTTTAACCGGCTACCTGGGAATTACGATAATTCCG
>asv2
ATTCGGACCTGGGAATTACGATAATTCCGCTTGAAGTGGGACTGAACTTTGAACG
>asv3
CTAAGTAACCTGGAACTTAGGGGTAAAAGTCCGTCAACTGAGAGAGTCGAGAACG
...
```

This fasta file is not always provided, but is critical to fully benefit from the possibilities of DNA data.

DNA files to Darwin Core

The examples on the previous page show how DNA-derived data contains multiple dimensions of information that is often recorded in separate interconnected tables.

In the Darwin Core archive format, each unique sequence by sample combination is considered **one occurrence**. Therefore the data from these tables will need to be formatted to the “long format”, including a row for each sequence in each sample. See the figure below for a demonstration of how this can be done.



Sample	asv	Date	Filter	Amount	Location	Depth	Replicate	Kingdom	Phylum	Class	Order	Family	Genus	Species	Sequence	OTU/ASV
sample1	asv1	2023-07-05	0.2	1200	site1	10	1	Eukaryota	Mollusca	Bivalvia	Ostreida	Ostreidae	Dendostrea	Dendostrea frons	ATTCGGGGT...	12560
sample1	asv2	2023-07-05	0.2	1200	site1	10	1	Eukaryota	Chordata	Actinopteri	Chaetodontiformes	Leiognathidae	Equulites		ATTCGGACC...	4
sample1	asv3	2023-07-05	0.2	1200	site1	10	1	Eukaryota	Chordata						CTAAGTAAC...	459
sample 2	asv1	2023-08-03	0.2	800	site2	10	1	Eukaryota	Mollusca	Bivalvia	Ostreida	Ostreidae	Dendostrea	Dendostrea frons	ATTCGGGGT...	123
sample 2	asv2	2023-08-03	0.2	800	site2	10	1	Eukaryota	Chordata	Actinopteri	Chaetodontiformes	Leiognathidae	Equulites		ATTCGGACC...	3
sample 2	asv3	2023-08-03	0.2	800	site2	10	1	Eukaryota	Chordata						CTAAGTAAC...	45

Once you have all your information in one place, we can begin data formatting as some columns will be placed in the Occurrence table, and others in the DNA Derived Data extension table.

Beginning DNA occurrence dataset formatting

As we begin the data formatting process, we will first fill in the Occurrence core table, and then complete the DNA Derived Data extension (as well as the eMoF extension, if applicable, for any measurements taken). Due to the high quantities of occurrence records from a DNA sample, the data formatting is usually done with scripts using, for example, R or Python. A few examples of this are available at (here links to the [monterey](#) and [uk](#) examples). Note that these scripts are now out of date in terms of how to deal with unknown sequences, however they are still a good example for how to automate data formatting for DNA data.

Let's look at formatting the Occurrence table first.

Format Occurrence core table

Formatting the Occurrence core table will generally follow the guidelines we learned in Modules 2 and 3. As you will recall, required terms for Occurrence core tables include:

- occurrenceID
- scientificName
- scientificNameID (strongly recommended)
- basisOfRecord
- occurrenceStatus
- eventDate
- decimalLatitude
- decimalLongitude

However, the following additional terms are strongly recommended to additionally include for DNA datasets. Please see the notes below for some important differences in term usage for DNA data.

- Class Occurrence | DwC: [organismQuantity](#) (**the regular DwC definition does not apply for DNA data, see below**)
- Class Occurrence | DwC: [organismQuantityType](#) (**the regular DwC definition does not apply for DNA data, see below**)
- Class Occurrence | DwC: [associatedSequences](#)
- Class Identification | DwC: [identificationRemarks](#)
- Class Identification | DwC: [identificationReferences](#)
- Class Identification | DwC: [verbatimIdentification](#)
- Class Taxon | DwC: [taxonConceptID](#)
- Class Event | DwC: [sampleSizeValue](#) (**note the regular DwC definition does not apply for DNA data, see below**)
- Class Event | DwC: [sampleSizeUnit](#) (**note the regular DwC definition does not apply for DNA data, see below**)
- Class Event | DwC: [samplingProtocol](#)

Here it is important to understand the meaning of the abundance values recorded in [organismQuantity](#) and [sampleSizeValue](#). As mentioned before, the quantities recorded with sequencing studies always represent relative abundance to the total reads in the sample, and cannot be directly compared across samples. This is due to the nature of the sample processing protocol and the amplification of DNA with PCR, which biases the original quantities. In the field [organismQuantity](#), you will record the amount of that unique sequence in that specific sample (i.e. 33 reads). In the field [sampleSizeValue](#), you will record the total number of all reads in that specific sample (i.e. 15310 reads). This information will allow the person accessing the data to calculate the relative abundance of that sequence in the sample. Both fields [organismQuantityType](#), and [sampleSizeUnit](#), get the value "DNA sequence reads", as it is of high importance that sequence abundances are not confused with organism abundances recorded by traditional methods. The abundance information can usually be found in the "OTU-table".

[associatedSequences](#) should contain a reference to the URL domain where genetic sequence information associated with the Occurrence can be found, e.g. a link, identifier, or list (concatenated and separated) of identifiers. Can link to archived raw barcode reads and/or associated genome sequences, like a public repository. It is recommended that links contain the domain name (e.g. NCBI) in the URL, for example: <https://www.ncbi.nlm.nih.gov/bioproject/PRJNA887898/>. The actual sequence of the occurrence will be documented in the DNA Derived Data extension.

[identificationRemarks](#) should be used to record information on how the taxonomic information of the occurrence was reached against which reference database, and, if possible, with which confidence. For example "RDP annotation confidence: 0.96, against reference database: GTDB". This information should be recorded in the bioinformatic protocol of the study. Note: this information will also be recorded in the DNA derived extension in the fields [otu_seq_compappr](#) and [otu_db](#).

[identificationReferences](#) should include a link to the bioinformatic pipeline or publication where the identification process is explained in detail.

[taxonConceptID](#) should include the taxonomic ID of the sequence (non-Linnean). Often genetic sequences can be assigned an ID linked to a reference database that is not a Linnean name. These taxonomic names or IDs from other taxonomic databases (like the NCBI taxonomic database) can be recorded in this field. For example: NCBI:txid9771. The name linked to this ID can then be recorded in the field [verbatimIdentification](#). We will discuss recording unknown sequences later in this lesson.

`samplingProtocol` can contain free-text that briefly describes the methods used to obtain the sample, or a link to a protocol that is recorded elsewhere.

We will look at an example dataset that demonstrates usage of these terms in Lesson 5. Let's next look at guidelines for the DNA Derived Data extension.

Format DNA Derived Data extension table

As mentioned earlier in this module, the DNADerivedData extension is meant to capture information related to the sampled DNA, including sampling, processing, and other bioinformatic methods.

The following (free-text) terms are strongly recommended to include in the DNADerivedData extension for eDNA data. We have bolded some terms of particular interest, details on these are provided below.

- DNA Derived | **DwC: DNA_sequence**
- DNA Derived | DwC: sop
- DNA Derived | **DwC: target_gene**
- DNA Derived | DwC: target_subfragment
- DNA Derived | **DwC: pcr_primer_forward**
- DNA Derived | **DwC: pcr_primer_reverse**
- DNA Derived | DwC: pcr_primer_name_forward
- DNA Derived | DwC: pcr_primer_name_reverse
- DNA Derived | DwC: pcr_primer_reference
- DNA Derived | DwC: Pcr_cond
- DNA Derived | DwC: annealingTemp
- DNA Derived | DwC: annealinTempUnit
- DNA Derived | DwC: ampliconSize
- DNA Derived | **DwC: env_broad_scale**
- DNA Derived | **DwC: env_local_scale**
- DNA Derived | DwC: env_medium
- DNA Derived | DwC: lib_layout
- DNA Derived | DwC: seq_meth
- DNA Derived | DwC: otu_class_appr
- DNA Derived | DwC: otu_seq_comp_appr
- DNA Derived | DwC: otu_db

For a complete list of terms, as well as descriptions for those above, see the [DwC DNA Derived Data extension page](#).

The most important field in the extension is the **DNA_sequence** field, where the ASV/OTU sequence will be recorded. This field can then be searched with sequence alignment methods to, for example, find closely related sequences recorded in other studies, and will allow very powerful data comparison and analysis in the future.

The remaining metadata fields will help the person accessing the data to filter out the data of interest (e.g. specific genetic region with **target_gene**, **target_subfragment**, or **pcr_primer** fields), link to the public sequence databases with the MixS specific fields (e.g. **env_** fields), and evaluate the reliability of the sequence annotation method (e.g. **otu_** fields).

It is recommended to use [Environment Ontology \(ENVO\)'s biome classes](#) to describe the environmental system from which the sample was extracted. Like other identifiers, it is important to provide the exact ENVO reference identifier. Environmental systems are described in the two fields **env_broad_scale** and **env_local_scale**. The **env_broad_scale** is meant to provide a coarse resolution for which environment your sample came from. For data being published to OBIS, it is likely that this will be [marine biome \(ENVO:00000447\)](#). In general, we recommend selecting a biome subclass from ENVO. Then, of course, for the local scale you should identify the specific environment your sample was obtained from (e.g., coastal water, benthic zone, etc.). See the screenshot below for an example of ENVO's marine biomes. You may have to search by keywords for your specific biome.

Class Hierarchy

```
Thing
+ entity
+ continuant
+ independent continuant
+ material entity
+ system
+ environmental system
+ ecosystem
+ biome
+ aquatic biome
+ freshwater biome
- marine biome
- estuarine biome
- marine salt marsh biome
+ marine pelagic biome
+ marine benthic biome
- epeiric sea biome
+ marginal sea biome
+ mediterranean sea biome
- ocean biome
+ marine upwelling biome
```

Next of course, is the extendedMeasurementOrFact extension.

Format eMoF table

If you have taken any measurements related to the specimen, the environment, or related to the sampling protocol that aren't DNA-specific, please record this information in the extendedMeasurementorFact (eMoF) extension table. Environmental measurements could include water temperature and salinity.

For eMoF table guidelines, please see [Module 3 Lesson 4](#).

As we have learned, including taxonomic information is important to ensure our datasets are interpretable. But naming our DNA sequences isn't always so straight-forward, especially if we have uncharacterized sequences in our datasets. Let's discuss that next.

Taxonomic sequence names: known or unknown sequences

The taxonomic names and link to LSIDs from WoRMS are added in the same way as any other dataset (see guidelines for populating `scientificNameID`, Module 2). However, it is good to note that with genetic data, the names can be less defined, i.e. often containing subscripts like “unknown species”, “environmental sample” etc. As mentioned, the OBIS database keeps Linnaean names as the standard; allowing easy comparison and search by species names. To search for the LSIDs of your names you can use the R packages `worms::wm_records_names`, `taxize::get_wormsid`, and `taxize::get_boldid`. As a reminder from Module 2, names that are not well defined should be added at a higher taxonomic level. The original annotated name can still be included in the field `verbatimIdentification`.

Dealing with unknown sequences

It is important to understand the significance of unknown and uncharacterized sequences in genetic studies. Sequences are given taxonomic names based on comparisons to a reference database. The reference databases contain sequences that have been submitted with a name. Ideally, the reference database is a collection of sequences that are derived from vouchered, morphologically identified specimens. Notably, currently this is often not the case and sequences can also have erratic annotations. Furthermore, only a small portion of species have sequences in reference databases. Due to this reason, typically many sequences in any given study will remain uncharacterized. This is especially the case for tropical regions with high biodiversity. By recording all sequences, including uncharacterized sequences, we make sure that the information is not lost, even if the annotation is currently incorrect or missing. These uncharacterized sequences can then still be compared to other studies, and can be given a taxonomic name as more specimens are sequenced and added to the reference databases.

For unknown sequences it is required to populate the `scientificName` field with “Incertae sedis”, or the lowest taxonomic information if available. For example, if it is only known which Class a sequence belongs to, populate `scientificName` with the associated Class name. Similarly, `scientificNameID` should be populated with the WoRMS LSID for the name given to `scientificName`. For records recorded as Incertae sedis, `scientificNameID` should be populated with **urn:lsid:marinespecies.org:taxname:12**. We recommend also populating `verbatimIdentification` with the name that was originally documented (e.g. phototrophic eukaryote).

We have learned what fields are required and strongly recommended when formatting DNA data belonging to Categories I and II. We better understand the Occurrence core + DNA derived data extension + (optionally) extendedMeasurementOrFact extension structure required for DNA datasets.

Before we look at an example that implements these guidelines, let's learn how to format data belonging to Category III, qPCR data.

Introduction to compiling qPCR data

Compiling qPCR data is a little bit different than compiling eDNA or metabarcoding data. One of the main differences is that there are **no sequences recorded in the DNA_sequence field** of the DNA derived data extension. Instead, occurrences are based on detections made using species-specific primers and either qPCR (Quantitative Polymerase Chain Reaction) or ddPCR (Droplet-Digital Polymerase Chain Reaction), no sequencing is done. Both of these methods measure the quantity of a marker in a sample. Usually, a standard curve is added to each qPCR run to calculate the absolute copy numbers from the results. qPCR is based on a fluorescence signal that is emitted whenever a new copy of the target genetic region is made. Like in normal PCR, multiple rounds of amplification are done, and the amplification round where the fluorescence signal exceeds the threshold value (quantification cycle or C_q-value) is used to calculate the original concentration of the gene in the sample based on a standard curve.

It is very important to document the methods used for this type of data because the results can be sensitive to the specificity of the primers/assays used. Therefore documenting as much detail on the methodologies is important to ensure data interpretability.

Let's now go over some important aspects to consider while formatting this type of data, beginning with the Occurrence core.

As usual, when formatting the Occurrence table we will need to populate the required terms:

- occurrenceID
- scientificName
- scientificNameID (strongly recommended)
- basisOfRecord
- occurrenceStatus
- eventDate
- decimalLatitude
- decimalLongitude

Additional terms we strongly recommend including are:

- Class Occurrence | DwC:[recordedBy](#)
- Class Occurrence | DwC: [organismQuantity](#) (different from categories I and II, see below)
- Class Occurrence | DwC: [organismQuantityType](#) (different from categories I and II, see below)
- Class Event | DwC: [sampleSizeValue](#) (different from categories I and II, see below)
- Class Event | DwC: [sampleSizeUnit](#) (different from categories I and II, see below)
- Class Event | DwC: [samplingProtocol](#)
- Class Material Sample | DwC:[materialSampleID](#)

In the case of ddPCR, [organismQuantity](#) refers to the number of positive droplets/chambers in the sample, and [organismQuantityType](#) is the partition type (e.g., ddPCR droplets, dPCR chambers). [sampleSizeValue](#) will be populated with the number of accepted partitions, e.g. meaning accepted droplets in ddPCR or chambers in dPCR. [sampleSizeUnit](#) is the partition type, which should be the same as [organismQuantityType](#). All four of these fields are particularly important to include for ddPCR data.

In case of qPCR, these fields can be used for recording e.g. the number of copies that were calculated for the target gene in the sample. In this case [organismQuantityType](#) needs to contain the exact type of the measurement reported in the results. The field accepts any string, but the best practice would be to add a URI pointing to a vocabulary, as is done in the [extendedMeasurementOrFact](#) extension. The terms [sampleSizeValue](#) and [sampleSizeUnit](#) would not be used in this case.

[materialSampleID](#) should contain an identifier for the MaterialSample (i.e. occurrence record), rather than a digital record of the material sample. If an ID was obtained from a nucleotide archive, use the associated biosample ID. Otherwise, construct a persistent unique identifier from a combination of elements in the data that will make the [materialSampleID](#) globally unique, similar to how we learned to construct eventIDs and occurrenceIDs.

[recordedBy](#) can be populated with the names of the people, groups, or organizations responsible for recording the original Occurrence. You can use a concatenated list for multiple names, by separating values with a vertical bar (' | ').

Now let's move on to the DNA Derived data extension.

As we know, the DNADerivedData extension will capture important information related to the sampling, processing, and bioinformatic methods. For qPCR datasets, it is strongly recommended to document as much detail as possible, particularly details about the PCR primers used and the target gene. We recommend you include the following terms, where relevant. For term descriptions see [DNA derived data extension](#). We have again bolded some terms of particular interest.

Terms related to the sampling event:

- DNA Derived | DwC: env_broad_scale (same as categories I and II)
- DNA Derived | DwC: env_local_scale (same as categories I and II)
- DNA Derived | DwC: env_medium
- DNA Derived | DwC: samp_collect_device
- DNA Derived | DwC: samp_mat_process
- DNA Derived | DwC: samp_size
- DNA Derived | DwC: size_frac

Terms related to DNA and PCR methods:

- DNA Derived | DwC: sop
- DNA Derived | **DwC: concentration**
- DNA Derived | **DwC: concentrationUnit**
- DNA Derived | **DwC: methodDeterminationConcentrationAndRatios**
- DNA Derived | DwC: contaminationAssessment
- DNA Derived | DwC: target_gene
- DNA Derived | DwC: target_subfragment
- DNA Derived | DwC: ampliconSize
- DNA Derived | DwC: amplificationReactionVolume
- DNA Derived | DwC: amplificationReactionVolumeUnit
- DNA Derived | **DwC: baselineValue**
- DNA Derived | DwC: automaticBaselineValue
- DNA Derived | DwC: automaticThresholdQuantificationCycle
- DNA Derived | **DwC: thresholdQuantificationCycle**
- DNA Derived | DwC: pcr_analysis_software
- DNA Derived | DwC: pcr_primer_forward
- DNA Derived | DwC: pcr_primer_reverse
- DNA Derived | DwC: pcr_primer_name_forward
- DNA Derived | DwC: pcr_primer_name_reverse
- DNA Derived | DwC: pcr_primer_reference
- DNA Derived | DwC: pcr_cond
- DNA Derived | **DwC: pcr_primer_lod**
- DNA Derived | **DwC: pcr_primer_loq**
- DNA Derived | DwC: annealingTemp
- DNA Derived | DwC: annealinTempUnit
- DNA Derived | **DwC: probeQuencher**
- DNA Derived | **DwC: probeReporter**
- DNA Derived | **DwC: quantificationCycle**
- DNA Derived | **DwC: ratioOfAbsorbance260_230**
- DNA Derived | **DwC: ratioOfAbsorbance260_280**

There are many specialized qPCR terms that are possible to add to the dataset.

The terms `concentration`, `concentrationUnit`, `ratioOfAbsorbance260_230`, `ratioOfAbsorbance260_280`, and `methodDeterminationConcentrationAndRatios` are related to the original DNA sample before qPCR analysis, and can be useful for evaluating the prevalence of the marker in the sample as well as the purity of the DNA for any indication of PCR inhibition.

As with the metabarcoding dataset, the details of the PCR conditions and primers can be recorded in the multiple `pcr_` terms as well as `target_` terms, and `amplificationReactionVolume` and `amplificationReactionVolumeUnit`.

The main terms that are important for the quantification information and are different from the metabarcoding dataset are `baselineValue`, `thresholdQuantificationCycle` and `quantificationCycle`. The terms `pcr_primer_lod`, `pcr_primer_loq`, `probeQuencher`, `probeReporter` are additional terms specific for qPCR assays. The `baselineValue` indicates the number of cycles below which the signal is considered only background noise. The `quantificationCycle` is the most important and indicates at which cycle the particular sample crossed the detection threshold, this will be different for each sample. It is recommended to record this information, but not all of this may be easily available.

In this lesson we learned important distinctions for how to prepare qPCR or ddPCR data for publishing to OBIS. Unfortunately, very few such data has been mobilized to OBIS thus far. If you have qPCR data you'd like to publish to OBIS, we would be happy to help answer additional questions not addressed in this course.

To conclude this module, we will look at an example eDNA dataset to see how it aligns to DwC according to the previously discussed guidelines.

Category I and II example dataset: eDNA

For this lesson we will demonstrate how to format data according to the standards discussed previously, focusing on data that falls into either Category I or II with real datasets. The dataset is "[18S Monterey Bay Time Series: an eDNA data set from Monterey Bay, California, including years 2006, 2013 - 2016](#)". The data from this study originate from marine filtered seawater samples that have undergone metabarcoding of the 18S V9 region.

The formatting of this dataset is demonstrated in detail in the github repository: <https://github.com/iobis/dataset-edna>. A selection of samples from this collection were included in the publication "[Environmental DNA reveals seasonal shifts and potential interactions in a marine community](#)" which was published with open access in Nature Communications in 2020.

As we know, genetic datasets are constructed around an Occurrence core table, so let's start with that.

The first step is to populate the Occurrence core with all the required and highly recommended fields, as well as considering the eDNA and DNA specific fields. The Occurrence core contains the taxonomic identification of each ASV (Amplicon Sequence Variant) observed.

We will document the number of reads, as well as relevant metadata including the sample collection location, references for the identification procedure, and links to archived sequences.

As we know from Module 3, `occurrenceID` and `basisOfRecord` are some of the required Occurrence core terms. The `occurrenceID` which was built from elements of the data, and because we are dealing with DNA material, the `basisOfRecord` will be `MaterialSample`.

We additionally want to include the highly recommended fields `organismQuantity`, `organismQuantityType`, `sampleSizeValue`, and `sampleSizeUnit`. These fields contain the number of DNA sequence reads. In metabarcoding data, recall that the difference between `organismQuantity` and `sampleSizeValue` is that `organismQuantity` **refers to the number of reads of the sequence in the sample**. `sampleSizeValue` is the **number of total reads in the sample**. It is important to include both values because you can calculate the relative abundance of a sequence within the total sample.

You can see that a reference to the project and its sequences is provided in `associatedSequences`, making it easily accessible.

A selection of samples from this sample/plate were included in another publication (Djurhuus et al., 2020), which is recorded in `identificationReferences` along with the GitHub repository where the data can be found.

(Note: tables below are split for simplified viewing)

occurrenceID	basisOfRecord	organismQuantity	OrganismQuantityType	associatedSequences
11216c01_12_edna_1_S_occ1	MaterialSample	19312	DNA sequence reads	NCBI BioProject acc. nr. PRJNA433203
11216c01_12_edna_2_S_occ1	MaterialSample	16491	DNA sequence reads	NCBI BioProject acc. nr. PRJNA433203
11216c01_12_edna_3_S_occ1	MaterialSample	21670	DNA sequence reads	NCBI BioProject acc. nr. PRJNA433203

sampleSizeValue	sampleSizeUnit	identificationReferences	identificationRemarks
147220	DNA sequence reads	GitHub repository Djurhuus et al. 2020	Genbank nr Release 221 September 20 2017
121419	DNA sequence reads	GitHub repository Djurhuus et al. 2020	Genbank nr Release 221 September 20 2017
161525	DNA sequence reads	GitHub repository Djurhuus et al. 2020	Genbank nr Release 221 September 20 2017

scientificName	scientificNameID	kingdom	phylum	class	order	family	genus
Paracalanus	urn:lsid:marinespecies.org:taxname:104196	Eukaryota	Arthropoda	Hexanauplia	Calanoida	Paracalanidae	Paracalanus
Paracalanus	urn:lsid:marinespecies.org:taxname:104196	Eukaryota	Arthropoda	Hexanauplia	Calanoida	Paracalanidae	Paracalanus
Paracalanus	urn:lsid:marinespecies.org:taxname:104196	Eukaryota	Arthropoda	Hexanauplia	Calanoida	Paracalanidae	Paracalanus

Now let's look at how to create the DNADerivedData extension for this dataset.

Next, we can create the DNA Derived Data extension which will be connected to the Occurrence core with the use of `occurrenceID`. This extension contains the DNA sequences and relevant DNA metadata, including sequencing procedures, primers used and Standard Operating Procedures (SOPs).

We learned in Lesson 3 that `env_broad_scale` is populated with a biome term, and because we know samples from this dataset were taken from filtered seawater, we will choose [marine biome \(ENVO:00000447\)](#) for this field. For `env_local_scale`, samples were collected at a nearshore station, indicating we will want to search for a term related to the coast. We strongly recommend searching by keywords rather than attempting to search through the drop down menus in the tree. Searching for the keyword "coast", we see that the 7th result relates to coastal seawater, and variants thereof. This result is more specific than the first result, [coast \(ENVO:01000687\)](#), so we will select the term [coastal water \(ENVO:00001250\)](#) to populate `env_local_scale`.

The samples from this example dataset were collected by CTD rosette and filtered by a peristaltic pump system. Illumina MiSeq metabarcoding was applied for the `target_gene` 18S and the `target_subfragment`, V9 region. URLs are provided for all the protocols followed for nucleic acids extraction and amplification.

For a detailed description of the steps taken to process the data, including algorithms used, see the original publication. Adding Operational Taxonomic Unit (OTU) related data are highly recommended and should be as complete as possible.

The tables below show how to fill some of the recommended fields for the DNADerivedData extension.

<code>occurrenceID</code>	<code>env_broad_scale</code>	<code>env_local_scale</code>	<code>env_medium</code>
11216c01_12_edna_1_S_occ1	marine biome (ENVO:00000447)	coastal water (ENVO:00001250)	waterborne particulate matter (ENVO:01000436)
11216c01_12_edna_2_S_occ1	marine biome (ENVO:00000447)	coastal water (ENVO:00001250)	waterborne particulate matter (ENVO:01000436)
11216c01_12_edna_3_S_occ1	marine biome (ENVO:00000447)	coastal water (ENVO:00001250)	waterborne particulate matter (ENVO:01000436)

<code>samp_vol_we_dna_ext</code>	<code>nucl_acid_ext</code>	<code>nucl_acid_amp</code>	<code>lib_layout</code>	<code>target_gene</code>
1000ml	dx.doi.org/10.17504/protocols.io.xjufknw	dx.doi.org/10.17504/protocols.io.n2vdge6	paired	18S
1000ml	dx.doi.org/10.17504/protocols.io.xjufknw	dx.doi.org/10.17504/protocols.io.n2vdge6	paired	18S
1000ml	dx.doi.org/10.17504/protocols.io.xjufknw	dx.doi.org/10.17504/protocols.io.n2vdge6	paired	18S

<code>target_subfragment</code>	<code>seq_meth</code>	<code>otu_class_appr</code>	<code>otu_seq_comp_appr</code>
V9	Illumina MiSeq 2x250	dada2;1.14.0;ASV	blast;2.9.0+;80% identity;e-value cutoff: x MEGAN6;6.18.5;bitscore: 100 :2%
V9	Illumina MiSeq 2x250	dada2;1.14.0;ASV	blast;2.9.0+;80% identity;e-value cutoff: x MEGAN6;6.18.5;bitscore: 100 :2%

V9	Illumina MiSeq 2x250	dada2;1.14.0;ASV	blast;2.9.0+;80% identity;e-value cutoff: x MEGAN6;6.18.5;bitscore: 100 :2%
----	----------------------------	------------------	--

otu_db	sop	DNA_sequence
Genbank nr;221	dx.doi.org/10.17504/protocols.io.xjufknw or GitHub repository	GCTACTACCGATT...
Genbank nr;221	dx.doi.org/10.17504/protocols.io.xjufknw or GitHub repository	GCTACTACCGATT...
Genbank nr;221	dx.doi.org/10.17504/protocols.io.xjufknw or GitHub repository	GCTACTACCGATT...

pcr_primer_forward	pcr_primer_reverse	pcr_primer_name_forward	pcr_primer_name_reverse	pcr_primer_reference
GTACACACCGCCCGTC	TGATCCTTCTGCAGGTTACCTAC	1391f	EukBr	Amaral-Zettler et al. 2009
GTACACACCGCCCGTC	TGATCCTTCTGCAGGTTACCTAC	1391f	EukBr	Amaral-Zettler et al. 2009
GTACACACCGCCCGTC	TGATCCTTCTGCAGGTTACCTAC	1391f	EukBr	Amaral-Zettler et al. 2009

Sometimes the information you're looking for may not be simply presented in easy-to-find tables or figures. You may have to read associated published papers for methodology if you are not the data provider. As an example, here we have highlighted where some of this information can usually be found in a publication.

protocol. Libraries were loaded on a standard MiSeq v2 flow cell and one sequencing run per genetic locus was performed in a 2×250 bp paired end format using a v2 500-cycle MiSeq reagent cartridge. For the 16S rRNA, 18S rRNA and COI genes the MiSeq runs were performed with a 10% PhiX174 spike in, while for 12S rRNA 20% PhiX174 was added. Custom sequencing primers were added to appropriate wells of the reagent cartridge. Base calling was done by Illumina Real Time Analysis (RTA) v1.18.54 and the output of RTA was demultiplexed and converted to FastQ format with Illumina Bcl2fastq v2.18.0.

Bioinformatics. Resulting sequences from the four libraries (16S rRNA, 18S rRNA, COI, and 12S rRNA) were processed through a modified version of the banzai pipeline Unix shell script⁴⁷. Paired-end reads were assembled and filtered with PEAR⁴⁸. Homopolymers were removed with grep and awk commands. Samples were concatenated, and tags were removed. Primers were removed with cutadapt (Martin, EMBnet) and singletons were removed. Operational taxonomic units (OTUs) were clustered with Swarm⁴⁹. Chimeras were removed with VSEARCH v1.8.0.

Taxonomic annotations for 16S rRNA were performed with GreenGenes 13.5 downloaded on December 17, 2016. Taxonomic annotations for 18S rRNA, COI and 12S rRNA were performed with the GenBank nr BLASTN database that was downloaded from NCBI on September 20, 2017. The max target sequence within the BLAST algorithm was interpreted according to Shah et al.⁵⁰. Annotations with >80% identities were retained (Supplementary Table 1). These annotations were then interpreted through MEGAN6, which only considered hits that had a bitscore of greater than 100 and were within the top 2% highest scoring hits per contig. The most recent common ancestors of these hits were subsequently determined.

In this lesson we reviewed an example dataset and how it aligns to DwC so that it can be published to OBIS. We recognize the growing importance of making genetic data accessible and usable for a variety of purposes, thus this module has introduced and familiarized us with genetic data and how to format it so it is more interpretable and interoperable.

You have completed Module 4! Please complete Exercise 4-1 and Quiz 4 before moving on to Module 5.

In the next module we will build on principals of interoperability and learn more details on how to find and select appropriate controlled vocabulary.

Module 5: Controlled vocabulary

Site: [OceanTeacher Global Academy](#)

Course: Contributing and publishing datasets to OBIS (self-paced)

Book: Module 5: Controlled vocabulary

Table of contents

Module 5

Lesson 1: Using controlled vocabulary

- Important measurement IDs for OBIS
- NERC Vocabulary Server (NVS)
- P01 Collection
- Lesson summary

Lesson 2: Populating measurementValueID, measurementUnitID, and measurementTypeID

- Populating measurementUnitID
- Populating measurementValueID
- Populating measurementTypeID
- Lesson summary

Lesson 3: Selecting measurement IDs for measurementTypeID

- Selecting P01 codes for biological measurements: part 1
- Biological measurements: part 2
- Chemical measurements: part 1
- Chemical measurements: part 2
- Physical measurements: part 1
- Physical measurements: part 2
- No suitable P01 codes found
- Lesson summary & Video Links

End of Module



Introduction

In this module you will learn more about controlled vocabulary, specifically vocabulary used to populate the measurement ID fields in the extendedMeasurementOrFact table.



Learning Outcomes

After successful completion of this module, you should be able to:

- Understand why controlled vocabularies are important
- Understand the concepts behind the structure of vocabulary codes
- Select the most appropriate measurement ID code for your data
- Be able to request creation of new codes when necessary



How to Proceed

To succeed in this Module, you need to successfully complete the following lessons and exercises:

- Lesson 1: Introduction to controlled vocabulary
- Lesson 2: Populating measurementValueID, measurementUnitID, and measurementTypeID
- Lesson 3: Select P01 codes for measurementTypeID
- [Exercise 5-1: Find measurementTypeIDs](#)

as well successfully complete Quiz 5 with a score of $\geq 80\%$

- [Quiz 5](#)

Controlled Vocabulary

We have already discussed the importance of using standards to format our data in this course. Now we will learn more about the controlled vocabularies you can use to help label the variables and measurements within your dataset.

Let's start by understanding why such labels and vocabularies are important.

As we have learned earlier in this course, the extendedMeasurementOrFact (eMoF) terms `measurementType`, `measurementValue`, and `measurementUnit` are completely unconstrained and can be populated with free text. While free text offers the advantage of capturing complex and unclassified information, there is inevitable semantic heterogeneity (e.g., of spelling, wording, or language) that becomes a challenge for effective data interoperability and analysis.

For example, if you were interested in finding all records related to weight measurements, you would have to try to account for all the different ways "weight" was recorded by data providers (weight, wgt, Weight, wet weight, dry weight, etc.).

Try it! Use the [OBIS Measurement Type search tool](#) to see the diversity of `measurementTypes` that exist across published datasets in OBIS. Note that any `measurementTypeIDs` listed in this tool are solely for consultation purposes. In some cases codes may have been incorrectly chosen for the associated `measurementType`. You should always choose `measurementTypeIDs` based on your own data and the guidelines in this module.

Important measurement IDs for OBIS

The 3 identifier terms `measurementTypeID`, `measurementValueID`, and `measurementUnitID` are used to standardize the measurement types, values and units used in the eMoF table.

These three terms should be populated using controlled vocabularies referenced using machine readable Unique Resource Identifiers (URIs). For OBIS, we recommend using the internationally recognized [NERC Vocabulary Server](#), developed by the British Oceanographic Data Centre (BODC). However do note that the search interface for this server **does not work like Google**, and you must be careful when searching with keywords. We will learn more about searching this server later in the module. The server can also be accessed through:

- SeaDataNet facet search <https://vocab.seadatanet.org/p01-facet-search>
 - We **strongly recommend using this search interface** when selecting vocabulary for `measurementTypeID`, unless you are comfortable with accessing the NERC Vocabulary Server
- NERC Vocabulary Server search https://www.bodc.ac.uk/resources/vocabularies/vocabulary_search/
- Semantic Model Vocabulary Builder https://www.bodc.ac.uk/resources/vocabularies/vocabulary_builder/

Controlled vocabularies are incredibly important to ensure data are interoperable - readable by both humans and machines and that the information is presented in an unambiguous manner. Vocabulary collections like NERC NVS2 compile vocabularies from different institutions, authorities, and communities (e.g., [ISO](#), [ICES](#), [EUNIS](#), [SeaDataNet](#)), allowing you to map your data to them. In this way, you could search for a single `measurementTypeID` and obtain all related records, regardless of differences in wording or language used in the data.

Let's learn more about the NERC Vocabulary Server (NVS).

The NERC Vocabulary Server (NVS)

Each vocabulary “term” in NVS is a *concept* that describes a specific idea or meaning. For consistency, we will refer to individual vocabularies in NVS as **concepts**. Concepts within NVS are organized into collections that group concepts with commonalities (e.g. all concepts pertaining to units).

Sometimes collections contain concepts that are deprecated. Terms can be deprecated due to duplication of concepts, or when a term becomes obsolete. You should not use any deprecated concepts for any measurement ID, however you do not need to update old datasets if a concept used has since become deprecated. Deprecated concepts can be identified from lists on NVS because their identifier will have a red warning symbol, and the page for the term itself will indicate the concept is deprecated in red lettering (see below). Deprecated concepts will also have a **replacedBy** concept that could be read automatically by software. Unfortunately, there is currently no notification system in place to automatically warn you if a previously used concept has become deprecated. It is good practice to occasionally confirm the concepts you or your institution use are still available for use.

Identifier ↑ **Concept** **DEPRECATED**
z302002M **Abundance of *Calanus helgolandicus* (ITIS: 85276: WoRMS 104466) [Stage: copepodites C5 plus adults] per unit volume of the water body by optical microscopy**

One of the more important collections for OBIS is the [P01 collection](#), we'll talk about that next.

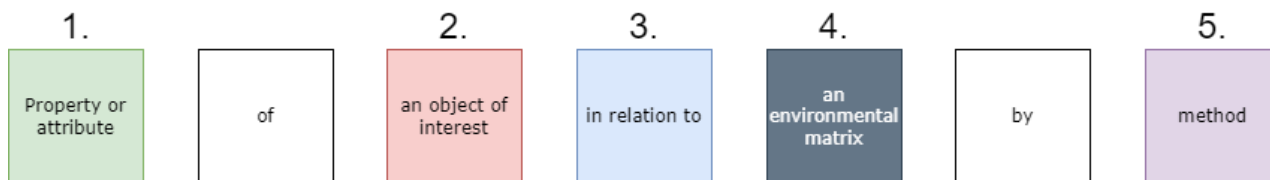
The P01 Collection

The [P01](#) is a large collection with >45,000 concepts. Each concept within this collection is composed of different elements that, together, construct a label you can use for a measurement type. Frequently we refer to concepts from the P01 collection as a “P01 code”. We use P01 codes to populate the `measurementTypeID` field.

A semantic model underlies each P01 code and the elements that compose them. There are 5 potential elements in this semantic model that, together, represent a specific concept to be used as a label. We will look at this model and each piece of it below.

1. **Property/attribute:** the measurement or observation of either an object of interest or a matrix, or both
2. **Object of interest:** a chemical object, a biological object, a physical phenomenon, or a material object
3. **In relation to:** how the measurement is related to the environment
4. **Environmental matrix:** what environment the measurement is in (e.g. water body, seabed); needed for most environmental measurements, but may not be necessary for e.g. biological measurements
5. **Method:** any specific methods used that are important to interpret the measurement

Not every element is required, but it is important to think about each piece of the model and how it may or may not apply to your measurement. We'll look at this in more detail when we learn how to populate the `measurementTypeID` field in Lessons 2 and 3.



In this lesson we learned why using measurement IDs is important, so that heterogeneities in the way data is recorded can be accounted for. We also learned about the semantic model underlying one of the bigger vocabulary collections important to OBIS, the P01 collection.

Next, let's look at how to populate `measurementUnitID`, `measurementValueID`, and the `measurementTypeID` fields in the next lesson.

Introduction to populating measurement ID fields

Selecting concepts to populate the `measurementUnitID` and `measurementValueID` fields tend to be more straightforward than populating `measurementTypeID`, so we will start this lesson with guidelines on how to select concepts for these two fields.

An important thing to remember is that when you populate any of these three measurement ID fields, the entire URI should be included. For example <http://vocab.nerc.ac.uk/collection/P01/current/SDBIOL01/> instead of just `SDBIOL01`.

Populating measurementUnitID

The `measurementUnitID` field is likely the easiest measurement ID field to populate. It is used to provide a URI for the unit associated with the value provided to `measurementValue` (e.g. cm, kg, kg/m²). This field **should be populated with concepts from the P06 collection**, titled "BODC-approved data storage units". Documentation for this collection is found: <https://github.com/nvs-vocabs/P06>.

The entire vocabulary list, including deprecated terms can be found: <http://vocab.nerc.ac.uk/collection/P06/current>, but you can **search for terms here**: https://www.bodc.ac.uk/resources/vocabularies/vocabulary_search/P06/. We strongly recommend using the second link to avoid potentially selecting deprecated terms.

Examples:

- Metres: <http://vocab.nerc.ac.uk/collection/P06/current/ULAA/>
- Days: <http://vocab.nerc.ac.uk/collection/P06/current/UTAA/>
- Metres per second: <http://vocab.nerc.ac.uk/collection/P06/current/UVAA/>
- Percent: <http://vocab.nerc.ac.uk/collection/P06/current/UPCT/>
- Milligrams per litre: <http://vocab.nerc.ac.uk/collection/P06/current/UMGL/>

Populating measurementValueID

The `measurementValueID` field is used to provide an identifying code for `measurementValues` that are **non-numerical** (e.g. sampling related, sex or life stage designation, etc.).

Note: it is not used for standardizing *numeric* measurements!

Unlike `measurementUnitID`, there is **more than one collection which may be used** to search for and use concepts from. The collection is dependent on which type of `measurementValue` you have. We provide some common, non-exhaustive examples in the table below.

Type of measurementValue	Collection	Collection Documentation	Complete Vocabulary List
Sex (gender)	S10	https://github.com/nvs-vocabs/S10	http://vocab.nerc.ac.uk/collection/S10/current/
Lifestage	S11	https://github.com/nvs-vocabs/S11	http://vocab.nerc.ac.uk/collection/S11/current/
Sampling instruments and sensors (SeaVoX Device Catalogue)	L22	https://github.com/nvs-vocabs/L22	http://vocab.nerc.ac.uk/collection/L22/current
Sampling instrument categories (SeaDataNet device categories)	L05	https://github.com/nvs-vocabs/L05	http://vocab.nerc.ac.uk/collection/L05/current
Vessels (ICES Platform Codes)	C17	-	http://vocab.nerc.ac.uk/collection/C17/current
European Nature Information System Level 3 Habitats	C35	-	https://vocab.nerc.ac.uk/collection/C35/current/

Behaviour descriptions or notes can be associated with an identifier from the [ICES Behaviour collection](#). See video below for details.

You can also populate this field with references to papers or manuals that document the sampling protocol used to obtain the measurement. To do this you should ensure the link provided is machine readable, so we recommend using:

- The DOI of the paper/manual
- The handle for publications on IOC's [Ocean Best Practices repository](#), (e.g. <http://hdl.handle.net/11329/304>)

A video walking you through examples of how to select vocabulary for `measurementValueID` is available below and through the OBIS YouTube Vocabulary Series, playlist here: <https://www.youtube.com/playlist?list=PLlgUwSvpCFS4hADB7Sif44V1KJauEU6UJ>. Some of the videos on this playlist are still under review by the OBIS Vocabulary team so be aware minor updates may occur in the coming months, including the video below.

Populating measurementTypeID

The `measurementTypeID` field should **only** be populated with codes from the [P01 collection](#).

As you learned in Lesson 1, P01 concept codes are complex in that they are based on a semantic model with several associated elements and/or sub-elements (see [P01 wheel](#) for example of how these elements make one concept together). Together, these elements unambiguously describe a measurement type.

It is important to remember that each element within a P01 code is meant to describe an aspect of the measurement: what is the measurement, what is the object or entity being measured, in what environment was the measurement taken, by what kind of methods, etc.? By taking together all these elements, we are able to have a unique and specific description to differentiate one measurement from another. More documentation about the P01 code and the semantic model it is based on can be found [here](#).

There are several ways of searching for a P01 code, for the purposes of this course, **we will be using the [SeaDataNet P01 Facet Search](#)**. You will learn how to use this search tool in Lesson 3. Keep in mind that its search function, like the NERC Vocabulary Server, works with keywords rather than free text search like we are accustomed to with Google. You may have to adjust your search approach when using the tool.

During searches for measurement types related to an occurrence, you may notice that specific taxonomic options are available to you, e.g., "abundance of Notommata". For OBIS, **all P01 codes should be generalized** - i.e. do *not* select species-specific codes. Instead, only **choose codes for "biological entities specified elsewhere"**! This is due to the Darwin Core Archive structure - taxonomic identification is already specified in the Occurrence table, but measurements are recorded in the ExtendedMeasurementOrFact table.

When you become comfortable with your understanding of P01 codes, you can also use the [BODC Vocabulary Builder](#) or simply search for terms directly on the [NERC Vocabulary Server](#).

In this lesson we learned which NVS collections can be used to populate the `measurementUnitID`, `measurementValueID`, and `measurementTypeID` fields.

Measurement ID	NERC Collection
<code>measurementUnitID</code>	P06
<code>measurementValueID</code>	S10, S11, L22, L05, C17, C35, ICES ...
<code>measurementTypeID</code>	P01

Next let's specifically focus on how to select P01 codes for `measurementTypeID`, because this field can be the most difficult to deal with.

Guidelines for measurementTypeID

As we have established, concepts from the P01 collection should be used to populate the `measurementTypeID` field.

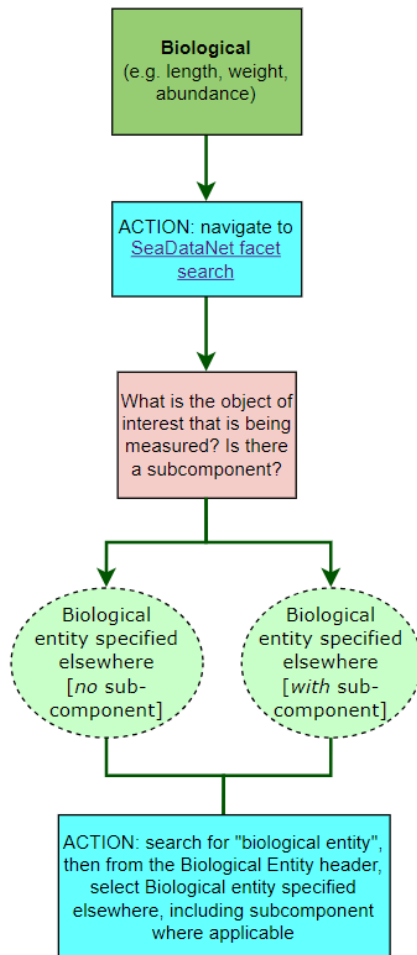
Because the semantic model P01 codes are based on contains five different elements to consider, we have created a decision tree to help guide you in selecting the best P01 code to represent your data. This decision tree should be used in conjunction with the [SeaDataNet Facet Search](#). The entire decision tree can be found in the [Resources for Exercises](#) section on the Course Overview as well as on the [OBIS Manual](#), but we will look at relevant pieces step by step.

We will focus on three types of measurements, biological, chemical, and physical measurements.

Selecting P01 codes for biological measurements: part 1

Biological measurements can include any measurement related to a biological entity, e.g. length, biomass, abundance, etc. As an example exercise, we will apply the guidelines below to the measurement “**abundance per unit area of the bed**”.

For biological measurements, after navigating to the [SeaDataNet Facet Search](#), the first step is to search for and/or select “biological entity specified elsewhere”. Let’s recall from the previous lesson that we **do not** want to choose P01 codes that are taxon specific (e.g. Wet weight biomass of *Hydrobia ventrosa*), because taxa are already identified in the Occurrence table.

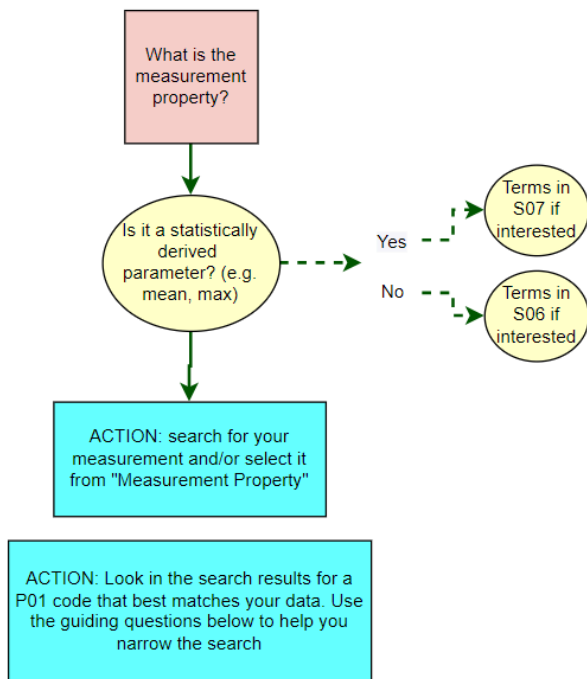


You can first use the Free Search box to search for “biological entity specified elsewhere”, and then from the Biological Entity section (you may have to scroll down), make sure you select “biological entity specified elsewhere” (pictured below). This will ensure all search results are directly relevant. For our example measurement, the measurement is not related to a sub-component of a biological entity (e.g. bell, mantle, central disc, etc.) so we will select “biological entity specified elsewhere”, not one that includes a sub-component.

BIOLOGICAL ENTITY (S25) ▼	
biological entity specified ...	(92)
biological entity specified ...	(1)
biological entity specified ...	(1)
biological entity specified ...	(1)
biological entity specified ...	(1)

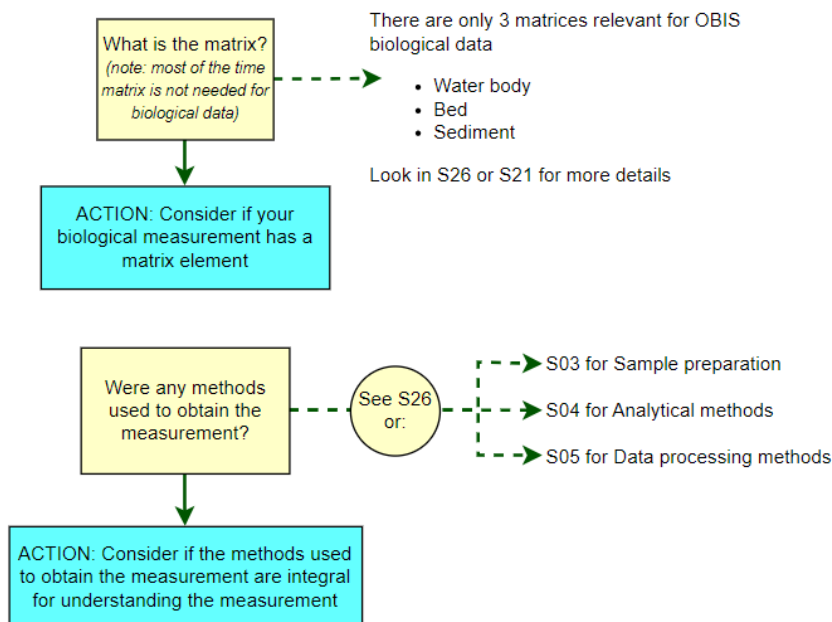
Biological measurements: part 2

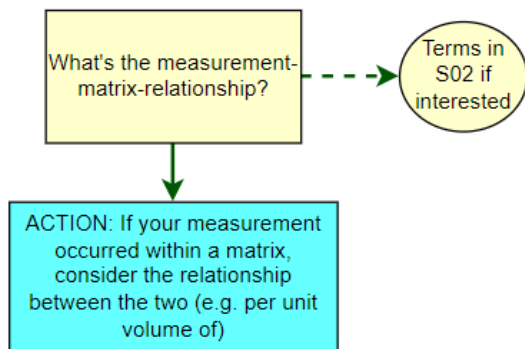
Once you have selected “biological entity specified elsewhere”, the next filter we want to apply is for our measurement property. One aspect here you want to consider simultaneously is if your measurement is statistically derived - is it an average, minimum, variance, etc.? Search by keyword for your measurement in the Free Search box or select it from the Measurement Property section to apply the filter.



For our example, we will select “Abundance” from Measurement Property. Now the search result list should be considerably shorter. There are only 5 results for our example measurement on abundance.

You may be able to already find the best P01 code at this step. However, depending on the complexity of your measurement you may need to consider the matrix (or environment, e.g. water body, bed), the methodology, and the relationship between the matrix and the measurement (see below sections of the decision tree). These aspects are not as frequently relevant for biological measurements in OBIS data, but it is important to still give thought if they are applicable to your measurement type. This could apply to measurements of organisms within, for example, a certain volume of the water body. As we learned already, methods are important to include in your `measurementTypeID` if they are integral for understanding the measurement.





In our example, the measurement is “abundance per unit area of the bed”. The matrix here is the bed. When we select “bed” from Matrices, only one result is left: <http://vocab.nerc.ac.uk/collection/P01/current/SDBIOL02/>. This matches the description of our measurement. It is not too specific, nor too generic.

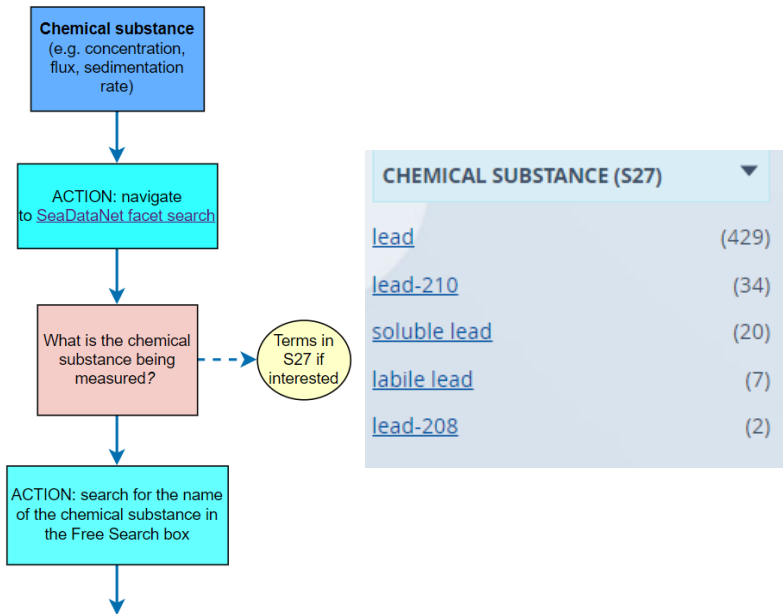
A few examples of codes that are more generic vs more complex are outlined in the table below. Note that it is possible to over filter and end up with a list of codes that do not fit your measurement. In these cases, you can remove filters to obtain a broader search result. We will look at an example of this situation when we review physical measurements.

Generic P01 code examples	Complex P01 code examples
Sex of biological entity specified elsewhere	Wet weight biomass of biological entity specified elsewhere per unit volume of the water body
Length (fork length) of biological entity specified elsewhere	Cell volume of biological entity specified elsewhere by cell size measurements and computation using stereometric formulas
Trophic status of biological entity specified elsewhere	Abundance of biological entity specified elsewhere per unit volume of the sediment
Count (in assayed sample) of biological entity specified elsewhere	Wet weight biomass of biological entity specified elsewhere per unit length sampled of the water body

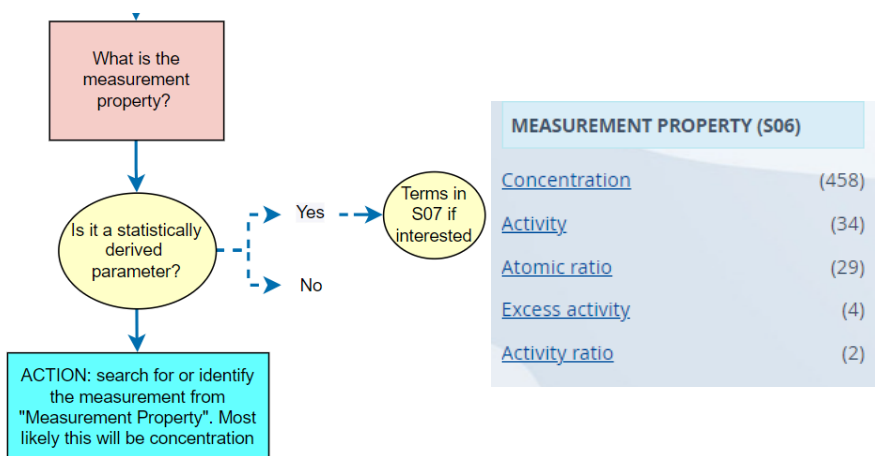
If you cannot find a P01 code that represents your data, you **can request one or leave the field blank and update the dataset later** when the applicable code(s) have been created. We will go over details for how to do this at the end of this lesson. For now, we will move on to guidelines for chemical measurements.

Chemical measurements: part 1

Identifying codes for chemical measurements starts a little differently from finding biological measurements. As an example, we will find a P01 code for the **concentration of lead within an organism**. After navigating to the [SeaDataNet Facet Search](#), we will first search for the chemical that is being measured using the Free Search box: lead. Also, select the chemical from the Chemical Substance section, similar to the procedure you followed during the biological measurement search.



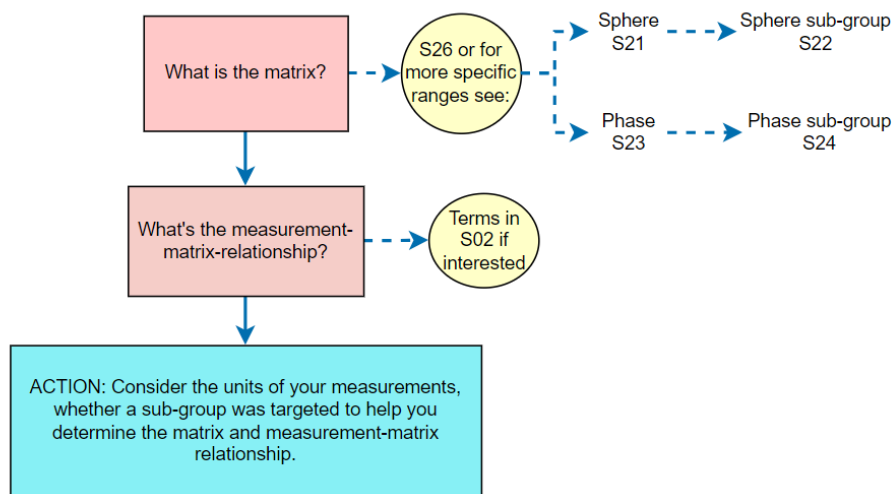
After this, you should select and/or search for the measurement property. In many cases for chemical measurements this will be concentration. Because concentration is available from Measurement Property, we will select it from here for our example. Otherwise we could search by keyword.



Chemical measurements: part 2

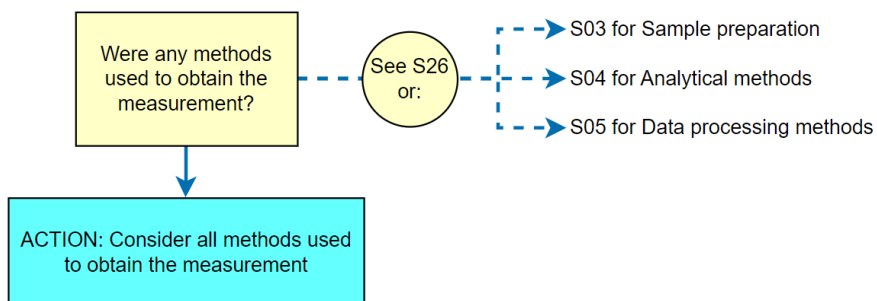
Next, consider the matrix for your measurement. Was it taken from a water sample (water body), from within an organism (biota [biological entity specified elsewhere]), or from the sediment? Here it is also important to consider whether your measurement represents a sub-sphere or sub-phase of the matrix. This would apply if you used a filter to target particulates < or > a particular size (e.g. water body [particulate 0.8-51um phase]).

Use keywords to search for your matrix, and then select the most appropriate matrix from the Matrices section to apply the filter. For our example, the concentration was within an organism - so the matrix would be biota. Immediately we notice that many of the codes listed are species-specific. We should now apply a filter to ensure we only have codes for "biological entity specified elsewhere" by adding these keywords to the Free Search. This brings the search results down from >300 to just one: <http://vocab.nerc.ac.uk/collection/P01/current/PBBIOTUK/> which represents our example measurement.



After specifying the matrix, the search will produce much fewer results. As the last step, consider the methods used to obtain the measurement. If applicable, either add keywords for this method to the Free Search, or manually look through the results list. Consider any applicable methods, including those used to prepare samples (e.g. filtration, acidification), analysis required to obtain the measurement (e.g. spectroscopy), or processing methods (e.g. normalization). Such methods can be integral for interpreting and comparing results. For example, the concentration of chlorophyll a can be determined by many different methods, including acetone extraction, high performance liquid chromatography, fluorometry, etc. These nuances are very important to capture in the P01 code you choose.

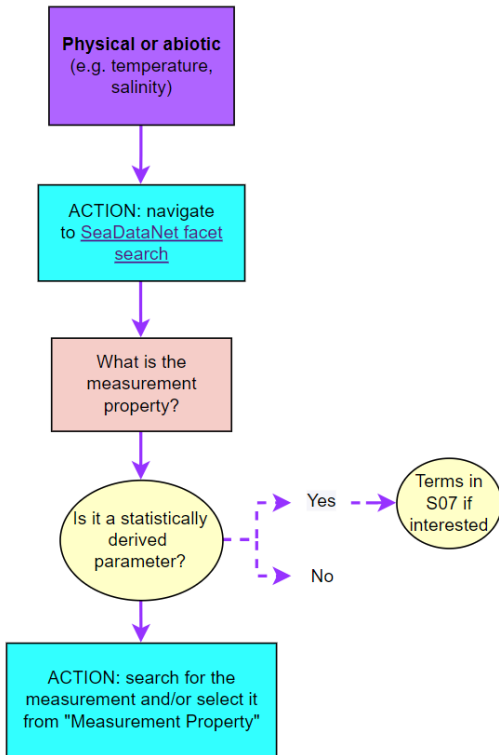
Unlike the guidance for the P01 codes applying to biological measurements, for chemical measurements you should choose a specific code with respect to the methods and matrix, rather than a more generic one. Note that currently there are limited P01 codes for chemical substances, so you may need to request code creation for your specific case. We will learn how to request terms (e.g. P01 codes) later in this lesson.



A brief reminder that **it is possible to over filter**. You may want to be cautious when applying specific filters, and be prepared to remove filters if search results become too narrow. We will see an example of this next, where we will look at the steps for physical measurements. These will share some similarities to those for chemical substances.

Physical measurements: part 1

Physical measurements include, but are not limited to, abiotic measurements frequently related to measuring the environment, e.g. water temperature, salinity. For an example exercise, we will use the **depth measured with an echo sounder** as the example measurement. Like before, use the [SeaDataNet Facet Search](#) and search by keywords in the Free Search box first. In this example, we will search for “depth”. As a reminder, make sure to also consider whether your measurement is statistically derived and use those as keywords when applicable (e.g. mean, standard deviation, maximum, etc.). Our example measurement is not statistically derived, so we will move on.



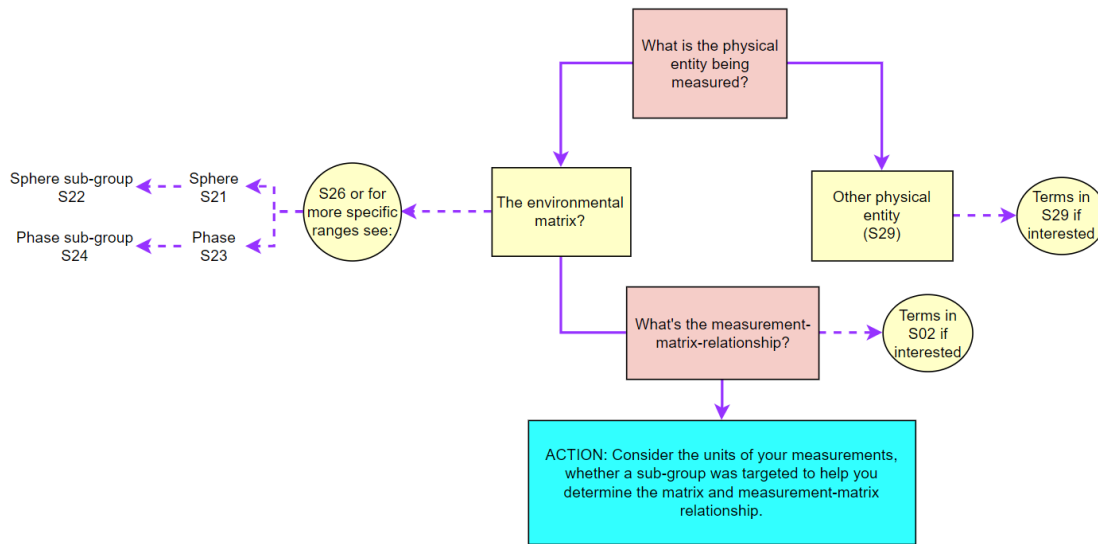
After searching for depth, we immediately notice a variety in the names of P01 codes related to depth. There is depth (spatial coordinate), sea-floor depth, depth, etc. If we select “Depth” from the Measurement Property, our search results get narrowed down to two: “Depth maximum of biological entity specified elsewhere on the bed by epibenthic sampling” and “Depth minimum of biological entity specified elsewhere on the bed by epibenthic sampling”. Neither of these represent our measurement! This is our first example of over filtering. Remove the “Depth” filter, and only keep the Free Search filter.

The screenshot shows the SeaDataNet search interface. The search term is "depth". The results are displayed in a table with columns for Conceptid and Preflabel. The table shows 109 results, with the first few rows visible.

Conceptid (109)	Preflabel
ADEPMP01	Depth (spatial coordinate) of Secchi disk relative to water surface in the water body by model prediction
ADEPW01	Depth (spatial coordinate) relative to water surface by barometric altimeter
ADEPZZ01	Depth (spatial coordinate) relative to water surface in the water body
BATHDPTH	Sea-floor depth (below mean sea level) (bathymetric depth)
BEDMXLTD	Mixed layer depth in the bed
CONEMXDP	Maximum penetration depth of cone in sediment by cone penetrometer
COREDIST	Depth (spatial coordinate) relative to bed surface in the bed
DBINAA01	Depth (spatial coordinate) of ADCP bin relative to water surface (bin depth) in the water body
DBINAW01	Depth (spatial coordinate) of ADCP bin relative to water surface (bin depth) in the water body by acoustic doppler wave array
DEPHCV01	Depth (spatial coordinate) relative to water surface in the water body by computation from probe free-fall time using unspecified algorithm

The next question to ask yourself is: what is the entity being measured? This is different from chemical measurements, because in many cases the matrix itself (i.e. water, air, sediment, biota, etc) is what is being measured. If this is the case, be sure to also consider if there are sub-groups to the matrix your measurement applies to (e.g. filtrate <2um,

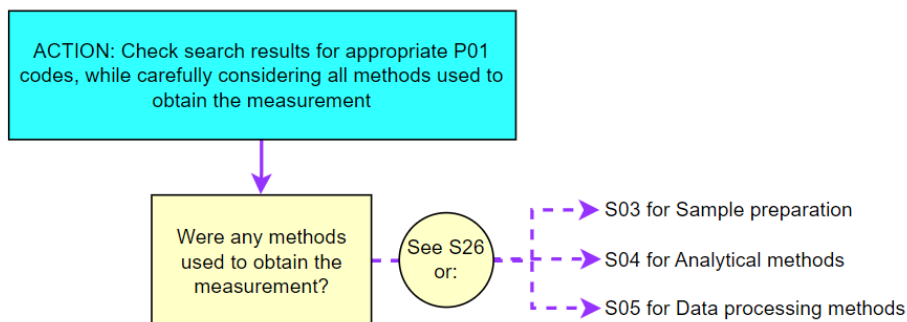
see [S22](#) and [S24](#) for more examples). If your measurement pertains to another physical entity, use this entity as a keyword. For a list of all potential physical entities, see the [S29 collection](#). Note you should not use any concepts from these other collections to populate the `measurementTypeID` field, they are provided here as an example of where you can find more information about the elements that compose a P01 code.



Then consider the relationship of the physical entity to the matrix. If the entity is the matrix, skip this step. To help you in this step, think about the units of your measurement; you can consult the [S02 collection](#) for examples of matrix-measurement relationships. Select appropriate matrices and measurement-matrix relationships from the facet search to help narrow down the results. In our example, the measurement is the depth of the water body - so the matrix itself is the entity being measured. We can select "water body" from Matrices.

Physical measurements: part 2

Next, you can manually read through results to identify P01 codes that represent your measurement. Be sure while doing so to consider the methods applied when you generated the measurement. For example, titration methods, calibration techniques, etc. As we learned when discussing chemical measurements, including methodologies in our P01 codes makes the measurements more easily comparable and understandable.



For our example, an echo sounder was used. Let's try using "echo sounder" as a search term. Now there are only 18 results, many of which include a correction to the measurement (e.g. Del Grosso, Kuwahara, etc.). Yet we cannot confirm if our example measurement used any correction methods, so we will not select any of those codes. This leaves us with two generic codes for echo sounders:

- [Sea-floor depth \(below unspecified datum\) {bathymetric depth} in the water body by echo sounder](#)
- [Sea-floor depth \(below instantaneous sea level\) {bathymetric depth} in the water body by echo sounder](#)

The first of these two codes has the following definition "*The distance from the seafloor to an unspecified representation of the sea surface elevation measured by echo sounder. Sound velocity corrections may or may not have been applied.*" This implies that there was some other representation of the sea surface.

The second code does not specify a definition, however, so we must interpret it based on the title. "*below instantaneous sea level*" refers to the depth below sea level at the moment it was taken (i.e. instantaneous). This code is the most generic while still being representative of our measurement, so we will select this one. This is a good example of codes that require some level of specificity, but not too much specificity - also demonstrating the caution you must take when selecting codes to ensure they accurately represent your measurement.

But what do you do if you reach the end of this process, for any type of measurement, and cannot find a P01 code that accurately represents your measurement?

If you cannot find a suitable code for your `measurementTypeID` (or `measurementValueID`, `measurementUnitID`), you can request a code to be created for you! Before doing so, make sure you have not over filtered the search results like we saw in the previous example. It can be worth removing some filters to first confirm there is no suitable code. Then to request new terms, submit a request to the [OBIS Vocabulary GitHub repository](#). Requests can also be emailed to vocab.services@bodc.ac.uk if you have a large number of vocabulary requests, or you cannot access GitHub. However, we strongly recommend you use GitHub if you can, as it allows for a dialogue about requests as required, longer-term documentation, and can be relevant for other users who may be interested in the same type of code creation. You may also register with the [BODC Vocabulary Builder](#) to request new terms. These requests should be based on combinations of existing concepts.

Note that you should **not** choose a vocabulary that only partially reflects your data. It is better to leave the field empty, and populate it later once a code is created than to use an incorrect code.

Finally, if you are unsure about whether a code fits your specific case, please feel free to ask questions on the aforementioned GitHub repository, the OBIS helpdesk email (helpdesk@obis.org), or the Vocab channel on the [OBIS Slack](#).

In this lesson you have learned how to choose measurementTypeIDs to describe your measurement data. We learned that, although choosing a P01 code can feel overwhelming due to the complexity of the underlying structure, breaking it down into steps helps us think about and identify the most appropriate P01 code for our situation. We also learned that if we are stuck or need to request a P01 code be created, we can do so using the OBIS Slack or Vocabulary GitHub Repository, respectively.

Practice using the guidelines in this module to select P01 codes for different measurements in Exercise 5-1. You may also watch the three videos below for more examples on how to select measurementTypeIDs for biological, physical, or chemical measurements. Each video provides 3 examples of varying complexity.

Biological measurements (available <https://www.youtube.com/watch?v=AfTh8UMxj7g&list=PLlgUwSvpCFS4hADB7SIf44V1KJauEU6UI&index=2>)

Chemical measurements (available https://www.youtube.com/watch?v=4cMM_WT9SeA&list=PLlgUwSvpCFS4hADB7SIf44V1KJauEU6UI&index=3):

Physical measurements (available <https://www.youtube.com/watch?v=S4-5OYpsU50&list=PLlgUwSvpCFS4hADB7SIf44V1KJauEU6UI&index=4>):

You have completed Module 5! After completing Exercise 5-1 and Quiz 5, move on to Module 6. In the next module we will learn how to run some quality control checks on our dataset, as well as how to address some common issues that come up during data formatting.

Module 6: Conducting Quality Control

Site: [OceanTeacher Global Academy](#)
Course: Contributing and publishing datasets to OBIS (self-paced)
Book: Module 6: Conducting Quality Control

Table of contents

Module 6

Lesson 1: Introduction to Quality Control

- Inspecting QC Flags
- Inspecting QC Flags with R
- Conducting QC
- obistools for QC
- Hmisc package for QC
- Lesson summary

Lesson 2: Resolving temporal uncertainties

- Date ranges
- Limited date information
- Text descriptions of dates
- Historical dates
- Lesson summary

Lesson 3: Resolving uncertain localities

- OBIS Maptool
- Getty Thesaurus & Google Maps
- Marine Regions Gazetteer tool
- Textual descriptions
- Lesson summary

Lesson 4: Uncertain taxonomic identification and measurement uncertainties

- Low confidence taxonomic identification
- Non-marine species
- Uncertain taxonomic measurements
- Lesson summary

End of Module



Introduction

You have already learned about data standards, and how to apply these standards to format datasets. In this module you will learn how to run quality control (QC) checks to verify any required or important step was not skipped and to help ensure our data is the best quality it can be before publishing to OBIS.



Learning Outcomes

After successful completion of this module, you should be able to:

- List potential quality flags, navigate to and understand QC flag reports, use R to inspect QC flags, identify the reasons why records are dropped from a dataset
- Run quality control checks on your own data using R
- Check published datasets for quality control flags using R packages
- Resolve uncertain temporal scope or uncertain eventDate
- Provide locality generalizations or estimates for missing latitude/longitude coordinates
- Resolve uncertainties related to taxa records, including when only fragments of organisms are recorded, or species marked as non-marine



How to Proceed

To succeed in this Module, you need to successfully complete the following lessons and exercises:

- Lesson 1: Introduction to Quality Control
- Lesson 2: Resolving temporal uncertainties
- Lesson 3: Resolving uncertain localities
- Lesson 4: Uncertain taxonomic identification and measurement uncertainties
- [Exercise 6-1: Practice georeferencing](#)
- [Exercise 6-2: Conduct QC checks](#)

as well successfully complete Quiz 6 with a score of $\geq 80\%$

- [Quiz 6](#)

Introduction to Quality Control

OBIS ignores records that do not meet a number of standards. For example, all species names need to be matched against an authoritative taxonomic register, such as the World Register of Marine Species. In addition, quality is checked against the OBIS-required fields as well as against any impossible values. OBIS checks, rejects, and reports the data quality back to OBIS nodes, but never changes records. A number of quality control (QC) tools have been developed to help you, such as a QC tool for species names and a QC tool for geography and data format

Why are records dropped?

How does OBIS determine which records are dropped and which are not? There are a number of potential reasons a record will be dropped entirely, including:

- The species is not marine
- The `scientificName` or `scientificNameID` did not match with WoRMS
- Issues with coordinates:
 - No coordinates given
 - `decimalLatitude` or `decimalLongitude` out of range
- The coordinate is zero

For each dataset published, a quality report is generated where the number of dropped records and other quality issues will be flagged. Some flags will only provide warnings and will not cause a record to be dropped. A complete list of QC flags can be found [here](#) but broadly speaking, potential flags relate to issues with location (coordinate format), event data (time, start and end dates), depth values being out of range, taxonomic issues (e.g. WoRMS name matching, non-marine or terrestrial species). QC reports can be found attached to individual datasets or when searching for data in OBIS. For example, if we searched for 'Crustacea' records, the following data quality report is given:

DATA QUALITY

🔊 DROPPED RECORDS

Dropped records	114,372	
> Not marine	53,459	
> Marine unsure	31,502	
> No coordinates	57,780	
> Zero coordinates	11,328	

You can see that >114,000 Crustacean records have been dropped, mostly due to records missing coordinates or species being flagged as non-marine. Because species are determined as being marine by WoRMS, sometimes species are marked as `not_marine` erroneously. We will return to this later in this module.

Generally, it's important to understand that records in your dataset may be rejected and not published (i.e. dropped) if the quality does not meet certain expectations. In other cases, quality flags are attached to individual occurrence records. To minimize the number of records dropped, be careful when formatting your data so that you are meeting the requirements. By running quality control checks before you publish, you can minimize the likelihood of records being dropped or flagged. Let's look a little more at the different QC flags that can be produced, and where to find flags in downloaded data.

Inspecting QC Flags in downloaded data

As we mentioned before, there are a number of QC flags that can tell you what sort of warnings are associated with a particular dataset or individual record. The QC checks OBIS implements as well as the associated flags are summarized [here](#).

There are several ways to inspect the QC flags associated with a specific dataset or any other subset of data. Data downloaded through the [OBIS mapper](#) or the [robis R package](#) will include a column named `flags` which contains a comma-separated list of flags for each record. There is also a data quality panel on the dataset pages that has a flag icon, which can be clicked to get an overview of all flags and the number of records affected. See below:

The screenshot shows a data quality panel with a callout box that says "Inspect quality flags" pointing to a flag icon. The panel is titled "DATA QUALITY" and contains two main sections: "DROPPED RECORDS" and "TAXONOMIC ISSUES". Each section has a table of flags and their counts, with progress bars on the right.

DATA QUALITY		
DROPPED RECORDS		
Dropped records	891	<div style="width: 100%;"></div>
> Not marine	85	<div style="width: 100%;"></div>
> No WoRMS match	501	<div style="width: 100%;"></div>
> No coordinates	243	<div style="width: 100%;"></div>
> Zero coordinates	0	<div style="width: 100%;"></div>
TAXONOMIC ISSUES		
Marine unsure	212	<div style="width: 100%;"></div>
No valid alternative	0	<div style="width: 100%;"></div>

Clicking this flag will open a table that lists the specific quality flags associated with the dataset, as well as any annotations added by the WoRMS annotated names list. When OBIS receives a scientific name that cannot be matched with WoRMS automatically, it is sent to the WoRMS team. The WoRMS team will then annotate the name to indicate if and how the name can be fixed. The picture below shows such a table with QC flags.

Quality flags

Quality flags are documented [here](#).

[report issue](#)

Flag	Records
depth_exceeds_bath	45,558
on_land	1,897
no_depth	1,123
no_match	501
no_accepted_name	438
no_coord	243
marine_unsure	212
not_marine	85
worms_annotation_todo	71
worms_annotation_unresolvable	68
lon_out_of_range	43

[previous](#) [next](#)

Clicking any of these flags in orange will take you to another table showing the specific affected records. For example, below is a list of records from a single dataset which have the [no_match](#) flag, indicating that no LSID or an invalid LSID was provided, and the name could not be matched with WoRMS. The column [originalScientificName](#) contains the problematic names, as [scientificName](#) is used for the matched name.

Occurrences

[report issue](#)

[open in mapper](#)

ID	scientificName	originalScientificName	scientificNameID
00090ad0-725b-49a2-872c-27318ac31718		Dactyliosolen flexuosus	
003b36ea-5888-4e00-9d9a-d9bfeaae3a78		Dactyliosolen laevis	http://www.algaebase.org/search/species/detail?species_id=kbfab0b5b9ff29246
00f641d6-2b2d-494f-acb0-4a33a812017c		Camptoplites bicornis magna	
0147999c-cd78-45f5-b29b-452febe260c1		Actinocyclus elegans	urn:lsid:marinespecies.org:taxname:
01e8207e-65e6-457b-8feb-b94e270448ca		Globigerina trilobata	urn:lsid:marinespecies.org:taxname:45814
028eb905-7018-4fdb-96f1-116a80844af0		Globigerina universa	
04865bed-a40f-4b92-9b01-d578521cb0f2		Synedra spathulata	
04e8d352-c0fc-4bca-b874-ad1a3021f144		Synedra spathulata	
05205a3c-69c4-408a-a332-31305ae67ba8		Globigerina trilobata	urn:lsid:marinespecies.org:taxname:45814
052e50f9-07d2-4901-956f-5111acf2692a		Chaetoceros chunii	urn:lsid:algaebase.org:taxname:87450
054ddaad-d930-4c27-b7e6-d9ef3b1ef937		Dactyliosolen laevis	http://www.algaebase.org/search/species/detail?species_id=kbfab0b5b9ff29246
055c4695-9bd2-4c2e-81c5-eafb4f6e28de		Pleurosigma smithianum	

At the top of the page, there's a button to open the occurrence records in the [OBIS mapper](#) where they can be downloaded as CSV. The occurrence table also has the [flags](#) column, so when inspecting non-matching names for example it's easy to check if the names at hand have any WoRMS annotations:

collectionCode	catalogNumber	dropped	flags
Discovery Reports	Stn WS 554 <i>Dactyliosolen flexuosus</i>	true	no_depth,no_match
Discovery Reports	Stn 578 <i>Dactyliosolen laevis</i>	true	no_depth,no_match
Discovery Reports	NAE No. 6 hole <i>Camptoplites bicornis magna</i>	true	no_match
Discovery Reports	Stn 666 <i>Actinocyclus elegans</i>	true	no_match,worms_annotation_reject_ambiguous
Discovery Reports	Stn WS 474 <i>Globigerina trilobata</i>	true	no_depth,no_match
Discovery Reports	Stn WS 221 <i>Globigerina universa</i>	true	no_match,worms_annotation_todo
Discovery Reports	Stn 577 <i>Synedra spathulata</i>	true	no_depth,no_match,worms_annotation_todo
Discovery Reports	Stn WS 545 <i>Synedra spathulata</i>	true	no_depth,no_match,worms_annotation_todo

Next let's look at how to inspect QC flags using R.

Inspecting QC flags with R

Inspecting flags using R is also very easy. The example below fetches the data from a single dataset, and lists the flags and the number of records affected. Notice that the `occurrence()` call has `dropped = TRUE` to make sure that any dropped records are included in the results:

```
library(robis)
library(tidyr)
library(dplyr)

# fetch all records for a dataset
df <- occurrence(datasetid = "f3d7798e-7bf2-4b85-8ed4-18f2c1849d7d", dropped = TRUE)
# unnest flags
df_long <- df %>%
  mutate(flags = strsplit(flags, ",")) %>%
  unnest(flags)
# get frequency per flag
data.frame(table(df_long$flags))
```

	Var1	Freq
1	depth_exceeds_bath	78
2	no_accepted_name	17
3	no_depth	5
4	no_match	138
5	not_marine	2
6	on_land	1
7	worms_annotation_await_editor	5
8	worms_annotation_reject_ambiguous	2
9	worms_annotation_reject_habitat	2
10	worms_annotation_todo	9
11	worms_annotation_unresolvable	7

So how can you help prevent or minimize the number of potential QC flags on your dataset? Let's take a look at the types of tools you can use to run QC checks.

There are many different tools you can use to help you perform quality checks on your data. These include:

- R package [obistools](#)
- R package and function [Hmisc::describe](#)
- ◦ Can give important summary statistics and identify numbers that don't match
- LifeWatch & EMODnet Biocheck
 - [Web UI](#) built on obistools. This tool requires your dataset to be published on an IPT (e.g., a test IPT such as <https://ipt.gbif.org/> where your dataset will not be harvested by GBIF or OBIS). Conducts additional checks that are not available in obistools.
 - [R package](#)
- [Lifewatch data services](#)
- The US Integrated Ocean Observing System [Standardizing Marine Bio Data Guide](#)
- [WoRMS taxon match tool](#)
- [GBIF data validator](#)
- [Python library for OBIS QC](#) developed by Canadian Integrated Ocean Observing System
- Excel Conditional Formatting tool - identify duplicated data
 - Excel > Home > Conditional Formatting > Highlight cells Rules > Duplicate values...

Because there are so many potential tools, let's look at the first two tools: obistools and Hmisc.

We have also developed a QC tutorial that can be used to assist you with Exercise 6-2, it demonstrates many of the QC steps outlined in this module and can be found at <https://shiny.obis.org/obislearn/>.

Running quality control checks with obistools

Within the `obistools` package, there are a number of useful functions you can use to check your dataset. Each function is designed to give you feedback on different elements of your data, and to report any inconsistencies.

Installing `obistools` requires the `devtools` package. Use the following code to install both packages:

```
install.packages("devtools")
```

```
devtools::install_github("iobis/obistools")
```

If you have difficulty installing `obistools`, please try updating your R packages, in particular the `vctrs` package. This can be done in RStudio in the Packages tab ("update" button) or by using the `update.packages()` command (you can choose which packages to update). If you cannot install `obistools` please reach out to helpdesk@obis.org and we will help you.

We recommend following this general procedure to start your QC checks:

1. Check that the **taxa match with WoRMS** using [obistools::match_taxa](#)

- o Example code

```
## this code conducts the taxon matching by telling the function to only match the unique species names in the
scientificName column
worms<-match_taxa(unique(occur$scientificName), ask=T)
## this piece of code merges the worms dataframe with the occurrence table (occur) by matching according to
scientificName
occur_match<-merge(occur, worms, by="scientificName", all= T)
```

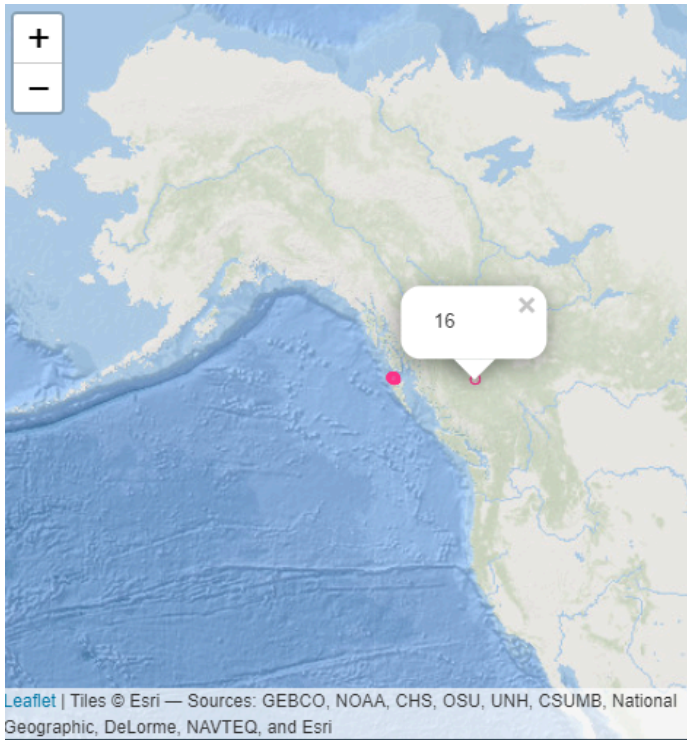
2. Check that **all required fields are present** in the Event and Occurrence tables with [obistools::check_fields](#)

Example:

```
> check_fields(event)
# A tibble: 55 x 4
  level field                row message
  <chr> <chr>                 <int> <chr>
1 error eventDate          NA Required field eventDate is missing
2 error scientificName      NA Required field scientificName is missing
3 error scientificNameID    NA Required field scientificNameID is missing
4 error occurrenceStatus    NA Required field occurrenceStatus is missing
5 error basisOfRecord       NA Required field basisOfRecord is missing
6 error decimalLongitude    1 Empty value for required field decimalLongitude
7 error decimalLongitude    2 Empty value for required field decimalLongitude
8 error decimalLongitude    3 Empty value for required field decimalLongitude
9 error decimalLongitude    4 Empty value for required field decimalLongitude
10 error decimalLongitude   5 Empty value for required field decimalLongitude
```

3. Check coordinates with [obistools::plot_map](#)

- o Plot them on a map to identify any points that appear outside the scope of the dataset. Using `obistools:plot_map_leaflet()` will additionally allow you to identify the row number for a particular data point.



- Check that **points are not on land** [obistools::check_onland](#)

The output of this function will return the rows in the dataframe that are on land, for example:

```
> check_onland(event)
  eventDate decimalLongitude decimalLatitude minimumDepthInMeters maximumDepthInMeters
16 2023-08-09T07:50:00      -123.9168         54.15019                0                12.20
43 2023-08-11T15:45:00      -132.6668         54.15012                0                24.99
44 2023-08-11T16:19:00      -132.6667         54.15012                0                26.76
```

- Ensure **depth ranges are valid** [obistools::check_depth](#)

You can specify a value to determine how much a given depth can deviate from the reference bathymetry layer to account for differences between the (potentially) coarser reference and the measurement taken in the real world.

```
> check_depth(event, report=T, depthmargin = 15)
# A tibble: 95 x 4
  level row field message
<chr> <int> <chr> <chr>
1 warning 14 maximumDepthInMeters Depth value (92.35) is greater than the value found in the bathymetry raster (depth=-1.0, margin=15)
2 warning 16 maximumDepthInMeters Depth value (12.2) is greater than the value found in the bathymetry raster (depth=-813.0, margin=15)
3 warning 28 maximumDepthInMeters Depth value (60.96) is greater than the value found in the bathymetry raster (depth=38.0, margin=15)
```

4. Check that the **eventID and parentEventID are present and corresponding** [obistools::check_eventids](#)
5. Ensure all **eventIDs in extensions have matching eventIDs in the core table** [obistools::check_extension_eventids](#)

```
> check_eventids(event)
# A tibble: 1 x 3
  field level message
<chr> <chr> <chr>
1 parentEventID error Field parentEventID is missing
> check_extension_eventids(event, occur)
  field level row
1 eventID error 2
message
1 eventID HaidaGwaii_2023_day2_AM07:20 has no corresponding eventID in the core
```

6. Check that **eventDate is formatted properly** [obistools::check_eventdate](#)

This may identify either rows that do not have any eventDate (e.g. a parentEvent that doesn't contain any date information)

or dates that were incorrectly formatted.

```
> check_eventdate(event)
# A tibble: 16 × 4
  level  row field      message
  <chr> <int> <chr>    <chr>
1 error     1 eventDate eventDate does not seem to be a valid date
2 error     2 eventDate eventDate does not seem to be a valid date
3 error     3 eventDate eventDate does not seem to be a valid date
4 error     4 eventDate eventDate does not seem to be a valid date
5 error     5 eventDate eventDate does not seem to be a valid date
6 error     6 eventDate eventDate does not seem to be a valid date
7 error     7 eventDate eventDate does not seem to be a valid date
8 error     8 eventDate eventDate does not seem to be a valid date
9 error     9 eventDate eventDate does not seem to be a valid date
10 error    10 eventDate eventDate does not seem to be a valid date
11 error    11 eventDate eventDate does not seem to be a valid date
12 error    12 eventDate eventDate does not seem to be a valid date
13 error    13 eventDate eventDate does not seem to be a valid date
14 error    26 eventDate eventDate 09-08-202316:20:00 does not seem...
15 error    46 eventDate eventDate 2023/08/12 does not seem to be a...
16 error    47 eventDate eventDate 2023/08/12 does not seem to be a...
```

Now let's look at how you can use the Hmisc package. This package will help us identify statistical outliers and other anomalies in the data.

Running quality control checks with Hmisc

The R package [Hmisc](#) has the function [describe](#) which can help you identify several discrepancies in your dataset.

It will summarize each of your variables for a given dataset. This can help you quickly identify any missing data and ensure the number of unique IDs is correct. For example, in an Occurrence table with 1000 records, there should be 1000 unique occurrenceIDs.

In the example output below we are checking an Occurrence data table. We can see that there are 407 records in total. There is only **one** unique CollectionCode, **27** unique eventIDs, and **407** unique occurrenceIDs. No records are missing any of these identifiers. From this output, it looks like the Occurrence table is okay. However, we would still have to check coordinates, formatting, etc. as we learned on the previous page.

```
library(Hmisc)
library(Hmisc)
data<-read.csv("example_data_occur.csv")
describe(data)

 12 Variables      407 Observations
-----
CollectionCode
  n missing distinct  value
 407      0         1 BIOFUN1

Value      BIOFUN1
Frequency    407
Proportion    1
-----
eventID
  n missing distinct
 407      0         27

lowest : BIOFUN1_BF1A01 BIOFUN1_BF1A02 BIOFUN1_BF1A03 BIOFUN1_BF1A04 BIOFUN1_BF1A05
highest: BIOFUN1_BF1M3  BIOFUN1_BF1M4  BIOFUN1_BF1M6  BIOFUN1_BF1M8  BIOFUN1_BF1M9
-----
occurrenceID
  n missing distinct
 407      0         407

lowest : CSIC_BIOFUN1_1  CSIC_BIOFUN1_10  CSIC_BIOFUN1_100 CSIC_BIOFUN1_101 CSIC_BIOFUN1_102
highest: CSIC_BIOFUN1_95 CSIC_BIOFUN1_96  CSIC_BIOFUN1_97  CSIC_BIOFUN1_98  CSIC_BIOFUN1_99
```

In this lesson we learned about the many tools available to conduct QC checks on our data, as well as why certain records might get dropped. As a reminder, we also learned about an online QC tutorial that can assist you with Exercise 6-2, it allows you to run R code for the QC steps outlined in this module and can be found at <https://shiny.obis.org/obislearn/>.

For a demonstration of how to run these QC checks in R, watch the following video (<https://youtu.be/sNzipC6-r90>)

07 How to conduct quality control checks in R for OBIS data



Play Video

An additional note on the `match_taxa` function from the video above:

We can add extra parameters to our functions when matching taxonomic names to WoRMS using R. By adding the `ask` parameter, we will ensure pop-up questions will appear during taxon matching in cases where there are multiple matches. In some cases, taxa there may not be multiple matches but the WoRMS list returned might have scientific names that have resolved typos, etc. Then when we want to merge the two lists, taxa with typos or other errors will not be matched to its corresponding WoRMS LSITD. This is where we can add the `"all=T"` to the merge function, which will add extra rows to the output, one for each row in the occurrence file that has no matching row in `worms`. Of course, we will then have to manually check the resulting list to see if any extra rows have been added. See the lines of code below for how to add these parameters. The example code presented earlier in this lesson included these additions to the code.

```
worms<-match_taxa(unique(occur$scientificName), ask=T)
occur_match<-merge(occur, worms, by="scientificName", all= T)
```

In the next few lessons we will learn how to handle some common QC issues that might arise, including uncertainties about location, date, taxonomic, and measurement data.

Uncertain temporal ranges

There are times when the `eventDate` or the temporal scope of a dataset is either in question, or provided in an invalid format (e.g. textual description). In this lesson we will learn how to resolve different issues related to the date and time of a dataset. We will look at:

- Providing date ranges
- Dealing with limited date information
- Text descriptions of dates
- Historical dates

Let's start by looking at date ranges.

Date ranges

If you are uncertain about the exact date an event took place, or the date was only provided as a range (e.g. March to June 1982), date ranges can be provided in the `eventDate` field to capture this uncertainty.

You should use ISO 8601 format for date ranges only if you are certain on the date range. Do not include a date range if you are making assumptions or guesses. However, notes about any appropriate assumptions or interpretations on date ranges can be documented in the `eventRemarks` field. Such assumptions could include interpretations of log books where the event range is known, but the exact date of sub-events or sampling are not documented.

Be careful when entering date ranges because the start and end are inclusive. For example, entering 1870/1875-08-04 is equivalent to *any* date between 1870 and 1875-08-04. You may use date ranges in this way to capture some level of uncertainty in when an event occurred. Be sure to always document the original date description in `verbatimEventDate`.

Examples

Provided date	ISO 8601 formatted date range
April to August 1982	1982-04/1982-08
Winter 1830-1831 (with notes about locality to infer season)	1830-11/1831-02
1 Sept - 15 Sept 2008	2008-09-01/2008-09-15
May 16, 1731, 6PM to May 17, 3AM	1731-05-16T18:00/1731-05-17T03:00

Now let's take a look at what to do if you have limited information about the `eventDate`.

Limited date information

If only parts of the date are known (e.g., year but not month and day), you may provide the date to `eventDate` in ISO 8601 format while excluding the unknown elements. It is good practice to also populate the `dwc:year`, `dwc:month` and `dwc:day` fields with known information.

Important note: Do not use zero to populate incomplete dates. Simply end the date with the known information (e.g., 2011-03 instead of 2011-03-00). Additionally, if the year is unknown, you should only populate the `dwc:month` and `dwc:day` fields because `eventDate` cannot be formatted to exclude year. In these cases, `eventDate` is not necessary to fill.

Examples

Known date elements	Formatted date
year: 2002	<code>eventDate</code> 2002 <code>year</code> 2002 <code>month</code> blank <code>day</code> blank
year: 2013 month: October	<code>eventDate</code> 2013-10 <code>year</code> 2013 <code>month</code> 10 <code>day</code> blank
month: February day: 12	<code>eventDate</code> blank <code>year</code> blank <code>month</code> 2 <code>day</code> 12
year: 1999 day: 20	<code>eventDate</code> blank <code>year</code> 1999 <code>month</code> blank <code>day</code> 20

Next, let's look at some examples for when date is only provided as a written description.

Text descriptions of dates

If the date in your dataset was provided as a **textual description** that can be accurately interpreted, include the text description in the `verbatimEventDate` field. Then provide the interpreted date in ISO 8601 format in the `eventDate` field. Be sure to document any other important information in `eventRemarks`.

Examples

Example description	Interpreted date
10th day of the 20 day cruise beginning 4 June 1951	1951-06-13
In the interval of the university examinations from May 18 to 27, 1938	1938-05-18/1938-05-27
Sunday May 16th 1824. At 3 PM saw a fish... got her killed about 1/2 after 5PM.	1824-05-16T15:00/1824-05-16T17:30

We will conclude this lesson by discussing how to handle historical dates, dates predating the 1500s.

Historical dates

There can be difficulties formatting dates from historical data for a number of reasons. Some of these difficulties arise from the change in calendar systems, from the [Julian calendar](#) to the currently used (by most countries) [Gregorian calendar](#) metric system. This change was implemented in 1582, so any datasets predating this year must be converted to the Julian calendar system. Additionally - there is [no year zero](#), only -1 and 1, where -1 is BCE (Before Common Era) and 1 is CE (Common Era).

For historical dates that do not conform to the ISO 8601 format and to accommodate challenges associated with them, the OBIS Historical Data Project Team has developed the following recommendations:

- Always populate `verbatimEventDate` with the originally documented date so that it can be preserved. Place converted dates that align to ISO 8601 in the `eventDate` field, and document the changes you made to the original in `eventRemarks`
- When the exact date is unknown, provide a date range, e.g. the period 21 November 1521 to 29 August 1612 records as 1521-11-21/1612-08-29
- For **archaeological data**, you can use a combination of terms from the Darwin Core class [GeologicalContext](#) and the [Chronometric Age Extension](#). GeologicalContext terms can be used to capture some information, however the Chronometric Age extension allows for better description of the time period (e.g. age, period, etc.), and links to the Event core table. For such records, `eventDate` would be populated with the date of collection. Note however that currently, the Chronometric Age extension **will not be aggregated** when publishing to OBIS, but the extension will be available when an individual dataset is downloaded.
- If the historical record contains uncertain or sensitive location information, generalize the location information using polygons or lines as described in the next Lesson

For historical data originating from old records, such as ship logs or other archival records, we understand there can be additional issues in interpreting and formatting data according to DwC standards. We have already reviewed some solutions you can use (date range, partial date information), but we understand that these issues can vary wildly. If you need further help with historical data formatting, we currently recommend [submitting a Github issue](#) to get assistance with such issues.

In this lesson we learned how to deal with some common challenges related to dates and time, specifically:

- How to provide a range of dates to capture uncertainty or a particular time period
- How to deal with partial date information
- How to handle text descriptions of dates
- How to deal with historical data

In the next lesson we will learn how to resolve ambiguities around geographic locations.

Uncertain spatial extent

Sometimes locality information can be difficult to interpret, especially if records originate from historical data with vague descriptions, or descriptions/names of areas that no longer exist. If your dataset is missing `decimalLongitude` and `decimalLatitude`, but the locality name is given, there are a number of approaches you can take.

We will go over four alternatives you can use to resolve geographic uncertainties and obtain georeferencing information, depending on the information available to you:

1. OBIS Maptool
2. Getty Thesaurus and/or Google Maps
3. Marine Regions Gazetteer
4. Tools for dealing with textual descriptions of locations

OBIS Maptool

One approach to resolving geographic ambiguity is to generalize the location using polygons or lines, allowing you to capture a geographic area or a transect. You can use the [OBIS Map Tool](#) to obtain a [Well-Known Text \(WKT\)](#) string for point, line, or polygon features, which will be placed in the `footprintWKT` field. WKT strings are representations of the shape of the location and are placed in the Darwin Core Location class `footprintWKT` field. This is particularly useful for tracks, transects, tows, trawls, habitat extent, or when the exact location is not known. You can also use a polygon to generalize sensitive locality information, for example if the exact location cannot be shared.

WKT strings can be created using the Map Tool's Generate WKT function. This Map Tool can also calculate a midpoint and a radius for line or polygon features, which can then be added to the `decimalLongitude`, `decimalLatitude`, and `coordinateUncertaintyInMeters` fields, respectively. Alternatively, you can use the `obistools::calculate_centroid` function to calculate the centroid and radius for WKT polygons. To visualize and share WKT strings, [this wktmap tool](#) developed by Pieter Provoost can be used. Within the OBIS Map Tool, you can also access the [Marine Regions Gazetteer](#) which will additionally help you find locations. We'll go over details about the Marine Regions Gazetteer later in this lesson.

Whenever you are using a WKT string, it is important to place the corresponding projection in the `footprintSRS` field. Note the accepted spatial reference system for OBIS is EPSG:4326 (WGS84). This is the spatial reference used by the OBIS Map Tool, Google Maps, and is commonly used in many other map services.

For a walkthrough on how to use the OBIS Map Tool, watch the video tutorial below (available at <https://youtu.be/XM23WEvE364>). This video covers estimating coordinates, using the line and polygon tool, and obtaining and exporting WKT strings.

14 How to use OBIS Maptool part 1- Points, polygons, & obtaining WK...



Using the Getty Thesaurus and Google Maps

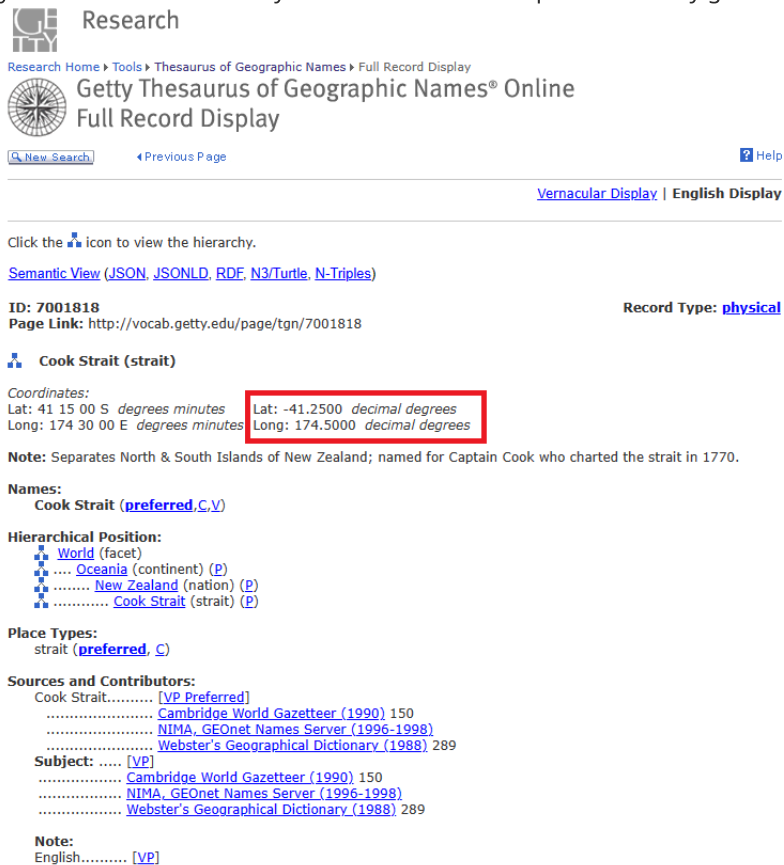
Using the [Getty Thesaurus of Geographic Names® Online](#) and [Google Maps](#) for georeferencing is relatively straight forward as long as you have the name of a locality to work with.

For an example, let's look at how we would georeference a locality documented only as "Cook Strait, New Zealand". The first step is to search for our location by name and nation (if relevant) in the search tool. There are convenient *Lookup* menus that can be used to help narrow down your search.

Getty Thesaurus of Geographic Names® Online



There is only one result for this search, and clicking on it will bring us to a page where we can obtain `decimalLatitude` and `decimalLongitude`. **Note:** Always be sure to fill in the `georeferenceSources` field to indicate the sources you used to obtain locality information! This is important for any georeferencing you do.




Research Home ▶ Tools ▶ Thesaurus of Geographic Names ▶ Full Record Display

Getty Thesaurus of Geographic Names® Online
Full Record Display

◀ Previous Page Help

[Vernacular Display](#) | [English Display](#)

Click the  icon to view the hierarchy.

[Semantic View \(JSON, JSONLD, RDF, N3/Turtle, N-Triples\)](#)

ID: 7001818 **Record Type:** [physical](#)
Page Link: <http://vocab.getty.edu/page/tgn/7001818>

Cook Strait (strait)

Coordinates:
Lat: 41 15 00 S *degrees minutes* Lat: -41.2500 *decimal degrees*
Long: 174 30 00 E *degrees minutes* Long: 174.5000 *decimal degrees*

Note: Separates North & South Islands of New Zealand; named for Captain Cook who charted the strait in 1770.

Names:
Cook Strait ([preferred, C, V](#))

Hierarchical Position:
World (facet)
..... Oceania (continent) (P)
..... New Zealand (nation) (P)
..... Cook Strait (strait) (P)

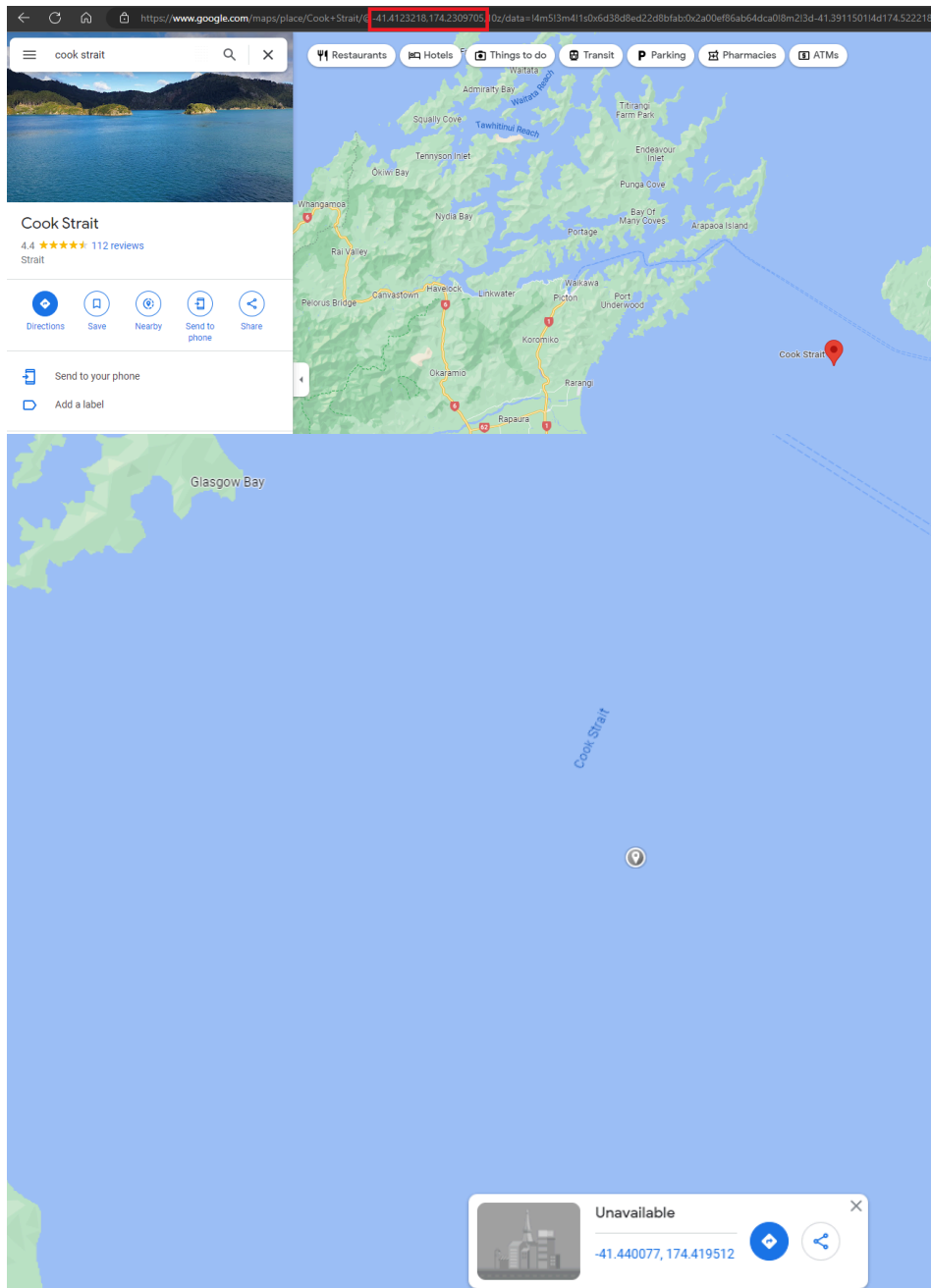
Place Types:
strait ([preferred, C](#))

Sources and Contributors:
Cook Strait..... [VP Preferred]
..... [Cambridge World Gazetteer \(1990\)](#) 150
..... [NIMA, GEOnet Names Server \(1996-1998\)](#)
..... [Webster's Geographical Dictionary \(1988\)](#) 289

Subject: [VP]
..... [Cambridge World Gazetteer \(1990\)](#) 150
..... [NIMA, GEOnet Names Server \(1996-1998\)](#)
..... [Webster's Geographical Dictionary \(1988\)](#) 289

Note:
English..... [VP]

For Google Maps, the coordinates can either be found in the url after searching for a location, or as a pop-up when you click on the map (see images below).



Note that whenever you record coordinates that were not the exact location (i.e. estimated using a thesaurus, a gazetteer or other methods like Google), you should always include georeferencing information in [georeferenceRemarks](#), as well as the [locationAccordingTo](#) field. The [georeferenceRemarks](#) field should be populated with a brief statement regarding any assumptions you made during georeferencing. [locationAccordingTo](#) will hold the name of the georeferencing tool you used, e.g. Getty Thesaurus of Geographic Names.

The Marine Regions Gazetteer

[Marine Regions](#) offers a [marine gazetteer search engine](#) to obtain geographic information and unique identifiers for marine regions. Once you have navigated to the [gazetteer search engine](#), you have two options to search by: enter the name of the desired locality, or enter an MRGID code.

Most likely you will have a locality name but not an MRGID. You may also select a [placetype](#) to search instead for types of regions that may be physical (e.g., seamount, bay, fjord, etc.) or administrative (e.g., exclusive economic zones, countries, etc.). You can specify specific sources if known (e.g., published paper, organization, etc.). Finally, you may also provide a latitude/longitude coordinate with a radius to obtain a list of regions near that point. A reminder that you should fill in the [georeferenceSources](#) field to indicate the source(s) you used to obtain locality information.


For example, let's look at the [IHO Bay of Fundy locality](#). Inspecting the page, there is a lot of useful information we can use. We can populate the following OBIS fields for our dataset, copying the information outlined in the red boxes from the figure below:

- (1) **locationID** from MRGID: <http://marineregions.org/mrgid/4289> (1)
- (2) **decimalLatitude** and **decimalLongitude** latitude and longitude coordinates of the location's midpoint in decimal degrees: 44.97985204, -65.80601556
- (3) **coordinateUncertaintyInMeters** precision: 196726 meters

Since we are obtaining all this locality data from Marine Regions, we must also populate the **locationAccordingTo** field. Here, we will provide the name of the gazetteer we used to obtain the coordinates for the locality - in this case you would write "Marine Regions". In **georeferenceRemarks** we must document that the coordinates are the region's midpoint, that locality information was inferred by geographic name, and, where applicable, place the original locality name in the field **verbatimLocality**. **georeferenceSources** will also be populated with the name of the gazetteer (Marine Regions in this case), and any other additional tools you used to complete the georeferencing.

Marine Gazetteer Placedetails

1. **MRGID** <http://marineregions.org/mrgid/4289>

Status Proposed standard 

Names

Language	Name	Source
----------	------	--------

English	Bay of Fundy	(1953). Limits of oceans and seas. 3rd edition. IHO Special Publication, 23. International Hydrographic Organization (IHO): Monaco. 38 pp. (look up in IMIS)
English	Fundy Bay	ASFA thesaurus

PlaceType IHO Sea Area

2. **Latitude** 44° 58' 47.5" N (44.97985204°)

Longitude 65° 48' 21.7" W (-65.80601556°)

3. **Precision** 196726 meter

Min. Lat 44° 5' 16.8" N (44.088°)

Min. Long 67° 19' 26.3" W (-67.324°)

Max. Lat 46° 12' 9.3" N (46.2026°)

Max. Long 63° 18' 17" W (-63.3047°)

Source (1953). Limits of oceans and seas. 3rd edition. IHO Special Publication, 23. International Hydrographic Organization (IHO): Monaco. 38 pp. (look up in [IMIS](#))

Relations

Part of	North Atlantic Ocean (IHO Sea Area)	[view hierarchy]
Adjacent to	Gulf of Maine (Gulf)	[view hierarchy]
Adjacent to	Maine (State)	[view hierarchy]
Adjacent to	Nova Scotia (Province (administrative))	[view hierarchy]

As mentioned earlier, you can also access the Marine Regions Gazetteer through the OBIS Maptool. The video below (available <https://youtu.be/CnefukzhnEM>) [demonstrates](#) using the embedded Marine Regions search function, as well as how to use the Gazetteer itself.

15 How to use the OBIS Maptool part 2 - Using the Marine Regions G...



Play Video

Text descriptions of locations

If you have data that only provides locality information in a text-format, you can try using the [GEOLocate Web Application](#). You can use this tool for one location at a time with the [Standard Client](#), or upload a CSV file for [batch processing](#). This tool lets you enter text descriptions in the “Locality String” field, and other relevant locality information (e.g. country, state, county) to obtain geographic coordinates.

Alternatively, you can use this [Biodiversity Enhanced Location Services](#) tool developed by VertNet. It can translate textual descriptions and provide `decimalLatitude`, `decimalLongitude`, `geodeticDatum`, and `coordinateUncertaintyInMeters` as a csv sent to an email address. For more information on this service, see the associated [GitHub](#).

Remember, OBIS is built on georeferenced biodiversity records, therefore, it is essential that the locality information in your dataset is accurate (or as precise as possible)!

In this lesson we learned about the different tools that can be used to georeference or generalize an ambiguous or sensitive locality. To practice what you have learned in this lesson, complete Exercise 6-1. When you are ready, move on to Lesson 4 which deals with low confidence taxonomic identification.

Uncertainties regarding taxonomic records

In this lesson we will learn how to deal with a variety of issues related to a taxonomic record. Such issues will include taxon identification that has low confidence, uncertainties regarding the count of sample organisms, and what to do if you encounter non-marine species.

Let's start with low confidence taxonomic information.

Low confidence taxonomic identification

For a variety of reasons, the taxonomic identification of a species may have low confidence. This includes when the scientific name contains qualifiers such as cf., ?, or aff. In these cases of low confidence taxonomic identifications, then you should:

- Put the name of the lowest possible taxon rank that can be determined with high-confidence in `scientificName` (usually Genus in these cases)
- Put qualifiers in `identificationQualifier` (e.g., cf., aff.)
- Put the species name in `specificEpithet`
- Place the rank of the taxon documented in `scientificName` (e.g., genus) in `taxonRank`
- Document any relevant comments in `taxonRemarks` or `identificationRemarks`

For example, let's say there was a specimen recorded as *Pterois* cf. *volitans*. The associated occurrence record would have the following taxonomic information:

- `scientificName` = Pterois
- `identificationQualifier` = cf. volitans
- `specificEpithet` = *leave blank*
- `scientificNameID` = the one for Pterois
- `taxonRank` = genus

If the provided genus name is unaccepted in WoRMS, it is okay to use the unaccepted name in this field. `scientificNameID` should contain the WoRMS LSID for the genus. There is a new Darwin Core term `verbatimIdentification` meant for containing the originally documented name, however this term is not yet implemented in OBIS so if you populate this field it will not be indexed alongside your data. Therefore you can use `originalNameUsage` to document original species names.

The use and definitions for additional Open Nomenclature (ON) signs (`identificationQualifier`) can be found in [Open Nomenclature in the biodiversity era](#), which provides examples for using the main Open Nomenclature qualifiers associated with physical specimens. Whereas the publication [Recommendations for the Standardisation of Open Taxonomic Nomenclature for Image-Based Identifications](#) provides examples and definitions for `identificationQualifier`s for non-physical specimens (image-based).

Changes in taxonomic classification

Taxonomic classification can change over time - so what does that mean for your datasets when records change classification?

Because OBIS relies on WoRMS as the taxonomic backbone, changes in taxonomic classification will be updated between a day to a few weeks from the date of change, unless triggered manually. This means that the WoRMS LSID associated with a species in question will be used to automatically populate the taxonomic classification with the updated information.

We recognize that there may be issues or update delays when larger changes occur (e.g. Family level splits). Then, records that are only identified to the Family level may not get updated properly. For example, there was a shift in coral taxonomy where the family Nephtheidae was split into Capnellidae and Alyconiidae for species occurring in the Northern Atlantic. While species that were identified as belonging to Nephtheidae have now been updated to belong to one of those two families, records that were only identified down to family (i.e., Nephtheidae) are still documented as Nephtheidae. Unfortunately, there is currently no solution for dealing with splits like this, aside from contacting data providers and asking them to change the taxonomy, or updating your dataset yourself to reflect the changes. We will learn how to update published datasets in Module 7.

Next let's look at what to do when faced with a flag that a taxon in your dataset is non-marine.

Non-marine species

If you encounter a flag or an error that a taxon is not marine, please confirm first whether the species is actually freshwater by cross referencing with [WoRMS](#) or [IRMNG](#). If the species is not marine (i.e. belongs to a non-marine genus), check with the data provider as necessary for possible misidentification. If the taxon was marked as non-marine in error, you can contact the WoRMS Data Management Team at info@marinespecies.org to discuss adding the taxon to the WoRMS register. You will be required to provide documentation in this case to confirm the marine status of the taxon. Otherwise, records marked as non-marine will be dropped from the published dataset, and this will be flagged in the data quality associated with your dataset.

Let's consider an example within [this dataset](#) on benthic macroalgae. Inspecting the data quality report we can see there are three dropped records due to species not being marine.

DATA QUALITY



DROPPED RECORDS

Dropped records	3	
> Not marine	3	
> No WoRMS match	0	
> No coordinates	0	
> Zero coordinates	0	

Clicking on the dropped records we can see which three species were dropped. By scrolling to the right of the table, we can see these records have two quality flags: NO_DEPTH and NOT_MARINE.

Occurrences

[report issue](#) [open in mapper](#)

imDepthInMeters	maximumDepthInMeters	occurrenceID	institutionCode	collectionCode	catalogNumber	dropped	flags
		FICOFLOAVZLA:Central:VAR:836900:8738				true	NO_DEPTH,NOT_MARINE
		FICOFLOAVZLA:Oriental:SUC:614549:9647				true	NO_DEPTH,NOT_MARINE
		FICOFLOAVZLA:Oriental:SUC:616768:9554				true	NO_DEPTH,NOT_MARINE

Let's take a look at the first species, *Pseudochantrasia venezuelensis*. When we search for this species on [WoRMS](#) we can see that the species is marked as freshwater only.

WoRMS taxon details

★ ***Pseudochantrasia venezuelensis* (L.G.D'Lacoste V & E.K.Ganesan) F.D.Ott, 2009**

AphiaID 836900 (um:lsid.marinespecies.org:taxname:836900)

Classification Biota > ★ Plantae (Kingdom) > ★ Biliphyta (Subkingdom) > ★ Rhodophyta (Phylum (Division)) > ★ Euhodophytina (Subphylum (Subdivision)) > ★ Florideophyceae (Class) > ★ Florideophyceae incertae sedis (Order) > ★ *Pseudochantrasia* (Genus) > ★ *Pseudochantrasia venezuelensis* (Species)

Status accepted

Rank Species

Parent ★ *Pseudochantrasia* F.Brand, 1897

Orig. name ★ *Rhodochorton venezuelense* L.G.D'Lacoste V & E.K.Ganesan, 1972

Synonymised names ★ *Audouinella venezuelensis* (D'Lacoste & Ganesan) Garbary, 1979 - unaccepted
★ *Rhodochorton venezuelense* L.G.D'Lacoste V & E.K.Ganesan, 1972 - unaccepted (synonym)

Environment marine, brackish, fresh

Original description Not documented

Taxonomic citation Guiry, M.D. & Guiry, G.M. (2023). AlgaeBase. World-wide electronic publication, National University of Ireland, Galway (taxonomic information republished from AlgaeBase with permission of M.D. Guiry). *Pseudochantrasia venezuelensis* (L.G.D'Lacoste V & E.K.Ganesan) F.D.Ott, 2009. Accessed through: World Register of Marine Species at: <https://marinespecies.org/aphia.php?p=taxdetails&id=836900> on 2023-03-23

Taxonomic edit history

Date	action	by
2015-03-31 10:06:03Z	created	Guiry, Michael D.
2015-06-26 12:00:51Z	changed	Guiry, Michael D.

Licensing Copyright notice: the information originating from AlgaeBase may not be downloaded or replicated by any means, without the written permission of the copyright owner (generally AlgaeBase). Fair usage of data in scientific publications is permitted.

[\[taxonomic tree\]](#)

Sources (1) Attributes (1) Links (3)

basis of record Guiry, M.D. & Guiry, G.M. (2022). AlgaeBase. World-wide electronic publication, National University of Ireland, Galway; searched on YYYY-MM-DD, available online at <http://www.algaebase.org> [details]

Cross-referencing with IRMNG, if we search for the genus, we can see that marine and brackish are stricken out, indicating the species is not marine.

IRMNG name details

***Pseudochantransia* F. Brand, 1897**

IRMNG_ID	1005149 (urn:lsid:irmng.org:taxname:1005149)		
Classification	Biota > Plantae (Kingdom) > Rhodophyta (Phylum) > Eurothodophylina (Subdivision) > Florideophyceae (Class) > Batrachospermales (Order) > Batrachospermales (Family) > <i>Pseudochantransia</i> (Genus)		
Status	uncertain > nomen dubium		
Rank	Genus		
Parent	Batrachospermales C.A. Agardh, 1824		
Direct children (11)	Species <i>Pseudochantransia lemnaeae</i> Brand, 1910 Species <i>Pseudochantransia thoreae</i> Brand, 1910 Species <i>Pseudochantransia tuomeyae</i> Brand, 1910 Species <i>Pseudochantransia beardstei</i> (Wolle) Brand, 1910 accepted as <i>Chantransia beardstei</i> Wolle, 1879 Species <i>Pseudochantransia boergesenii</i> F.D. Ott, 2009 accepted as <i>Audouinella parva</i> D.J. Garbary, 1987 Species <i>Pseudochantransia chalybea</i> (Roth) Brand, 1909 accepted as <i>Audouinella chalybea</i> (Roth) Bory de Saint-Vincent Species <i>Pseudochantransia hebdenii</i> (Kützting) Israelson, 1942 accepted as <i>Audouinella pygmaea</i> (Kützting) Weber-van Bosse, 1921 Species <i>Pseudochantransia macrospora</i> (Wood) Brand, 1910 accepted as <i>Audouinella macrospora</i> (Wood) Sheath & Burkholder, 1985 Species <i>Pseudochantransia parva</i> (D.J. Garbary) F.D. Ott, 2009 accepted as <i>Audouinella parva</i> D.J. Garbary, 1987 Species <i>Pseudochantransia pygmaea</i> (Kützting) Brand, 1909 accepted as <i>Audouinella pygmaea</i> (Kützting) Weber-van Bosse, 1921 Species <i>Pseudochantransia serpens</i> Israelson, 1942 accepted as <i>Audouinella serpens</i> (Israelson) Sheath ex Kumano, 2002		
Environment	marine, brackish		
Fossil range	recent only		
Original description	Not documented		
Descriptive notes	Taxonomic remark From Schneider & Wynne, 2007: A type species was not designated when this genus was validly published by Brand (1897)... O		
Taxonomic citation	IRMNG (2021). <i>Pseudochantransia</i> F. Brand, 1897. Accessed at: https://www.irmng.org/aphia.php?p=taxdetails&id=1005149 on 2023-03-23		
Taxonomic edit history	Date	action	by
	2007-02-14 23:00:00Z	created	Rees, Tony
	2011-12-31 23:00:00Z	changed	Morgan, Helen
	2016-11-22 09:43:47Z	changed	db_admin
	[taxonomic tree]		

Sources (6) Notes (2) Links (1)

basis of record SN2000 unverified/Dixon, 1982 [\[details\]](#)

basis of record Farr, E. R.; Zijlstra, G. (eds). (1996-current). Index Nominum Genericorum (ING). A compilation of generic names published for organisms covered by the ICN: International Code of Nomenclature for Algae, Fungi, and Plants. [previously: organisms covered by the International Code for Botanical Nomenclature] (2007 version). , available online at <https://naturalhistory2.si.edu/botany/ing/> [\[details\]](#)

If you have species that are marked as non-marine in these registers but are either supposed to be marine, or were found in a marine environment, then, as mentioned, you should contact WoRMS to discuss adding it to the register. For additions and/or edits to environmental or distribution records of a species, contact the WoRMS Data Management Team at info@marinespecies.org with your request along with your record or publication substantiating the addition/change.

Now let's consider cases where there is uncertainty regarding a measurement about an organism.

Uncertain taxonomic measurements

In some cases only fragments of organisms were found, or only a range of individuals is known. This can make it difficult to know how to properly populate the `individualCount` field.

In cases where only certain body parts (e.g., head, tail, arm) are recorded, you can incorporate this information in `measurementTypeID` by finding a code that documents sub-components of biological entities. If such a code for your sub-component does not exist, you can request the [creation of a P01 code](#). We will review more specifics about finding `measurementTypeID` codes in Module 5. If you are uncertain whether fragments are from the same individual, a range can be provided and the uncertainty can be recorded in `occurrenceRemarks`.

When providing a range to estimate the count, there are two [suggested](#) approaches:

- Use eMoF: place the data in the `extendedMeasurementOrFact` extension. This requires an additional two rows per Occurrence. One that will document the `minimumIndividualCount` and to document `maximumIndividualCount`.
- Place data into `dynamicProperties`: Include the information in the Occurrence record itself, with no extension, and instead document it in `dynamicProperties`, with a value such as `{"minimumIndividualCount":0, "maximumIndividualCount":5}`
 - Note that documenting the data in 'dynamicProperties' means the **information will not be machine readable** and may be more difficult for users to extract.

Abundance vs Count data: A brief clarification on abundance and count data: abundance is the number of individuals within an area or volume. This type of data is recorded in `organismQuantity`, where [organismQuantityType](#) should be used to specify the type of quantity (e.g., individuals, percent biomass, Braun Blanquet Scale, etc.). However, if the value is just the number of individuals without reference to a space (i.e. count), this information can be recorded in `individualCount`.

Using the `organismQuantity` and `organismQuantityType` fields is recommended in most cases as it allows you to record more details about the occurrence.

Finally, populate the `occurrenceRemarks` field to document all information about the decisions you made during this process.

In this lesson we learned how to deal with uncertainties about taxon occurrences - low confidence identification, questionable marine status, and uncertain organism quantity.

You have completed Module 6! Practice everything you have learned in this module by completing Exercise 6-1 and 6-2 to identify and correct as many quality control issues as possible.

In the next module we will learn how to publish our datasets. This step can be taken when you are sure the dataset will meet the quality control standards discussed in this module as best as possible. However, we also encourage you to publish data even if there are some things you need to update later (e.g. obtaining `measurementTypeID`s). You can always update datasets later with more data, or any changes that need to be made (e.g. change in taxonomy).

Module 7: Publishing datasets using the IPT

Site: [OceanTeacher Global Academy](#)

Course: Contributing and publishing datasets to OBIS (self-paced)

Book: Module 7: Publishing datasets using the IPT

Table of contents

Module 7

Lesson 1: Uploading data to the IPT

IPT

Creating a resource on the IPT

Uploading data to IPT

Mapping fields to DwC

Watch a video

Lesson 2: Populating IPT metadata and making the dataset public

Ecological Metadata Language

EML terms for datasets

Lesson summary

Lesson 3: Publishing on the IPT

Dataset versioning

Watch a video

Data Licenses

Lesson 4: Maintaining published datasets and publishing to GBIF

Updating datasets

Publishing to both OBIS and GBIF

Watch a video

End of Module

Introduction

In this module you will learn how to publish datasets to OBIS using an Integrated Publishing Toolkit (IPT).

Learning Outcomes

After successful completion of this module, you should be able to:

- Describe process for publishing data to OBIS
- Select the best data license for your dataset
- Locate the most appropriate IPT
- Create a resource in an IPT and publish it, download data files from IPT
- Provide metadata to publish dataset on IPT, update your dataset in the hosted IPT
- Understand responsibilities for managing an IPT
- Add DOI to your dataset, configure IPT to allow DOI assignment
- Publish a dataset with GBIF and OBIS at the same time,
- Describe difference between OBIS and GBIF, understand differences between requirements for publishing

How to Proceed

To succeed in this Module, you need to successfully complete the following lessons and exercises:

- Lesson 1: Uploading data to the IPT
- Lesson 2: Populating IPT metadata and making the dataset public
- Lesson 3: Publishing on the IPT
- Lesson 4: Maintaining published datasets and publishing to GBIF
- [Exercise 7-1: Fill in IPT metadata](#)

as well as successfully complete Quiz 7 with a score of $\geq 80\%$

- [Quiz 7](#)

Introduction to the IPT

Once you have finished formatting your data (Modules 2-4) and have conducted some basic quality control checks (Module 6), you are ready to publish your dataset to OBIS!

OBIS nodes can accept any data files from data providers, and these are published on their respective OBIS node Integrated Publishing Toolkit (IPT), after being formatted and checked. Data from [node IPTs](#) are then harvested by central OBIS to become accessible via the global database. The Integrated Publishing Toolkit (IPT) is developed and maintained by the Global Biodiversity Information Facility (GBIF). While GBIF maintains an [IPT manual](#), we will learn specific OBIS instructions in this module.

There are a few steps involved to publish a dataset, you must:

1. **Identify the most relevant IPT** for your [OBIS node](#) (you may have to contact your node manager to confirm IPT location)
2. Login to the IPT, or have the node manager create an account for you if you do not have one, so you can **upload your dataset(s)**
3. **Map each of your fields to Darwin Core** terms. This should be relatively straight forward if you have done this already during data formatting
4. **Fill all relevant metadata** to help users understand and cite your dataset, to assure data providers and institutions are properly cited
5. **Publish** and make your data public

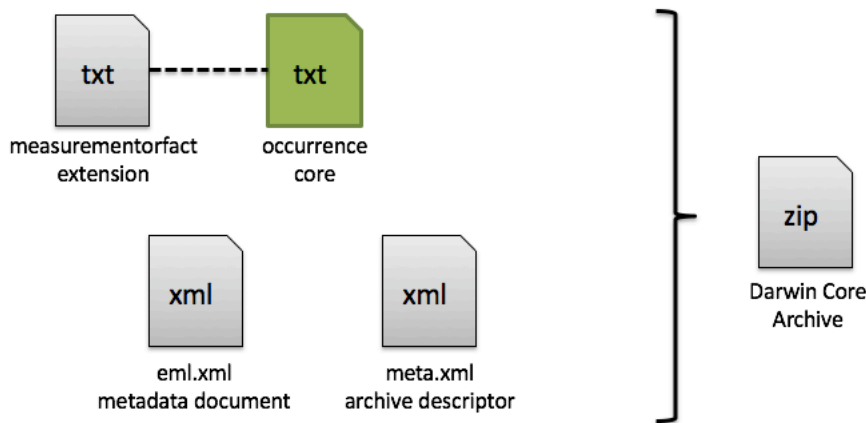
We will review details for each of these steps in the subsequent pages and lessons. Let's start by learning what an IPT is and how to use it.

Introducing the IPT

Biodiversity datasets and their metadata are published in OBIS using the Integrated Publishing Toolkit (IPT), developed by GBIF. The IPT is an open-source web application that can be customized by an OBIS node manager (if you are a node manager see OBIS manual for IPT admin details). An IPT-instance is used to publish and register all datasets. In general, the IPT software assists users in mapping data to valid Darwin Core terms, as well as archiving and compressing the Darwin Core content with:

1. A descriptor file: `meta.xml` that maps the core and extensions files to Darwin Core terms, and describes how the core and extensions files are linked
2. The `eml.xml` file, which contains the dataset metadata in [Ecological Metadata Language](#) (EML) format. We will learn about EML and metadata later in this module.

All these components (i.e., core file, extension files, descriptor file, and metadata file) become compressed together as a .zip file, and comprise the Darwin Core Archive.



To be able to create and manage your own dataset (called a “resource” by GBIF on the IPT), you will need a user account. Once you have determined which [OBIS node IPT](#) is suited for your dataset, you can contact your node manager to create an associated account for you. There will be a link on the sign-in page that will direct you to the IPT’s administrator to contact them. If your node’s IPT is not listed here, you will have to [contact the node manager](#) to get the link to their IPT.



INTEGRATED PUBLISHING TOOLKIT
free and open access to biodiversity data

Please log into your IPT account with your email address.

If you don't have an account yet, please ask the IPT administrator to create one for you.

Email

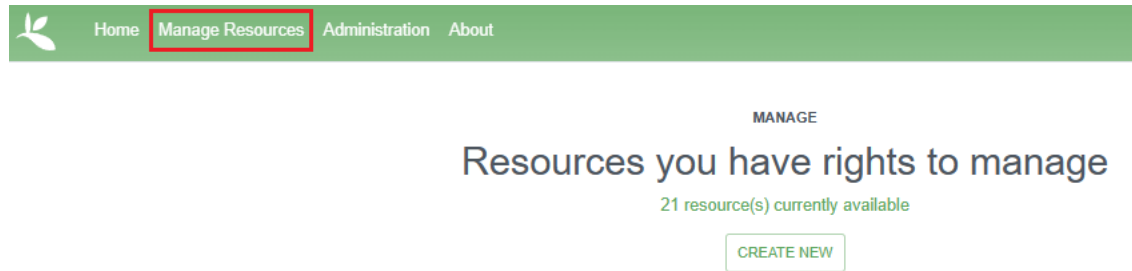
Password

LOGIN

Next we will learn how to create a resource on the IPT.

Creating a resource on the IPT

Once you have your account, log in at the top of the IPT page. Click on the tab Manage resources: it will display all the datasets you are managing and will be empty at first. You can create a new resource at the bottom of the page. The [GBIF IPT manual](#) has more detailed instructions, but we will learn the fundamentals here.



After clicking "Create New", the first thing that needs to be completed is the shortname of your resource. This shortname uniquely identifies your resource (i.e. dataset) and will eventually show up in the URL of this resource on IPT. These shortname identifiers are also used to create folders on the IPT and **they cannot be changed**. We therefore advise that the shortname:

- is unique, descriptive and short (max. 100 characters)
- does not contain a space, comma, accents or special characters

Some good examples of shortnames include VLIZ_benthos_NorthSea_2000 and UBC_algae_specimens.

Note: When you would delete a resource, please inform your node manager of this action! If you create a test file, please include `_test` at the end of your shortname.

Then select the type of data you are uploading: Occurrence, Checklist, Sampling-Event (i.e. Event Core), Metadata only, or Other. Note that [Checklist datasets](#) are accepted by GBIF, but not currently implemented in OBIS. However, you can still have checklist data hosted on OBIS IPTs.

You can also create an entirely new resource by uploading an existing archived resource. See the IPT manual section [Upload a DwC-A](#) for instructions.

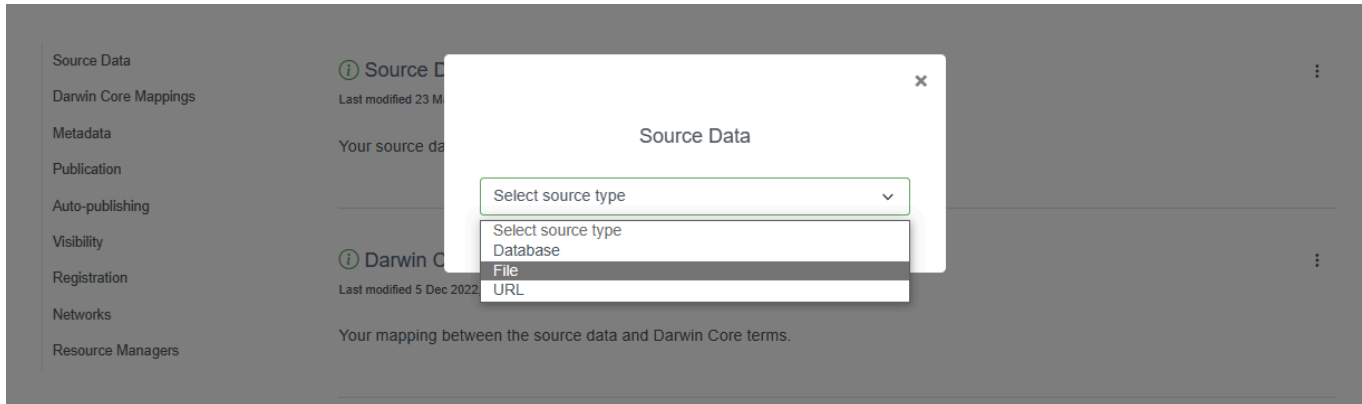
Please note the IPT has a 100MB file upload limit, however, there is no limit to the size of a Darwin Core Archive that the IPT can export/publish. Refer to the [File upload](#) section in the IPT manual, to find out how to work around the file upload limit.

Once you have created your resource, you will see an empty resource overview page. We next have to upload our data.

Uploading data to the IPT

Uploading your source file to the IPT is fortunately easy. From the Resource Overview page, under Source Data, click on Choose File. This is where you will select and add the files containing your Core table and (if applicable) extensions.

You might want to compress/zip your source file first to improve the upload speed of large files. The IPT will unzip them automatically once received. Follow the [IPT manual](#) for more detailed instructions (including the option to use multiple source files or to upload via a direct database connection). Accepted formats are delimited text files (csv, tab and files using any other delimiter), either directly or compressed as zip or gzip.



Let's follow an example dataset with an Event core. The first file we will upload will be a .csv file containing the Event core table. After it's been selected, we click "Add".

A source file detail page will then be shown, displaying how the IPT has interpreted our file (number of columns, rows, header rows, character encoding, delimiters, etc.). Here you can provide information about how the data table is encoded, how many header rows exist, the type of delimiters, and what type of character encoding you used. You may need to double-check the [character encoding](#) for your file, especially if you used any special characters (e.g., in species or place names). **UTF-8** is one of the most common encoding standards, and you can select this encoding when saving files, depending on the software used:

- **Windows: MS Excel:** select Save as. From the drop-down select the "CSV UTF-8 (Comma delimited)" option
- **Windows: Notepad:** Click on File, then Save as
 - In the drop-down menu "Save as type" drop-down, select All Files
 - In the "Encoding" drop-down on the bottom right, select UTF-8.
 - Be sure to name your file using the .csv extension (e.g., data.csv).
- **Mac: Numbers:** Select File then Export to... -> CSV.
 - Click Advanced options
 - Click the drop-down menu next to Text Encoding and select Unicode (UTF-8)
 - Click Next, then finally name your file, select a location, and click Save

Source Data

test dataset_elizabeth

SAVE

ANALYSE

PREVIEW

DELETE

CANCEL

Readable	● Yes
File	/data/training/resources/test_elizabeth/sources/example_data_event.txt
Columns	12
Rows	28
Size	11.9 KB
Modified	23 March 2023, 22:03:02
Source log	Download

① Source Name

example_data_event

① Number of Header Rows

1

① Field Delimiter

.

① Field Quotes

"


① Multi-value Delimiter

① Character Encoding

UTF-8

① Date Format

YYYY-MM-DD

 GBIF Integrated Publishing Toolkit (IPT) Version 2.7.2

Once you have specified all the above information, click the preview button to verify everything is correct, click anywhere on the screen to exit the preview, then click save. This will redirect you back to the Overview page for your dataset. You can add multiple files for each of your data tables by clicking the three vertical dots to the right and "+ Add". We will add a .csv file for an Occurrence extension, as well as one for the eMoF extension.

Source Data

Last modified 16 Feb 2023, 20:30:44

Your source data files and SQL sources for generating a Darwin Core Archive.

example_data_emof
134.8 KB | 2,221 rows/26 columns | 8 Feb 2023, 18:45:08

example_data_occur
71.2 KB | 999 rows/12 columns | 8 Feb 2023, 18:44:35

example_data_event
39.5 KB | 999 rows/27 columns | 8 Feb 2023, 18:38:57

Next we will map our data fields to Darwin Core in the IPT.

Mapping data columns to Darwin Core on the IPT

Because biodiversity data are published in the [Darwin Core](#) standard, we must map our data fields to the Darwin Core (DwC) standards and vocabulary. As we have mentioned earlier in this course, the DwC standard includes a list of defined terms and allows your data to be understood and used by others. It also allows an aggregator like OBIS or GBIF to digitally integrate your data with other datasets. DwC mapping is the process of linking the fields in your resource file with the appropriate Darwin Core terms. We learned that can be one of the most challenging steps in publishing data for two reasons:

1. The list of DwC terms can be overwhelming, so it might be difficult to select the ones that are appropriate for your dataset
2. The IPT currently only allows one-to-one mapping of fields, so the ease of mapping will depend on your database structure and on the feasibility of exporting as close to DwC standards as possible. You can contact your node manager or the OBIS secretariat at info@iobis.org to help guide you through the steps, review your mapping, suggest terms etc. You are also welcome to post questions in the [OBIS Slack](#).

To add the DwC mappings, click the three vertical dots to the right of this section and select "+ Add".

Darwin Core Mappings

Last modified 5 Dec 2022, 21:06:39

Your mapping between the source data and Darwin Core terms.



+ Add

A popup window will allow you to select your Core type to facilitate mapping to Occurrence or Event core. In this example, we will select Darwin Core Event for the Event core table. After confirming which table you'd like to map to, press Save.

This will bring you to the DwC mapping page. If you have already named several of your fields with DwC terms, they will have been auto-mapped. It is important to review each DwC class (Record, Event, Location, etc.) and ensure any mapped and unmapped fields are mapped to the correct DwC term. When you select a term, a drop-down menu will appear where you can select the appropriate field from your dataset. It is good practice to double-check that the auto-mapped fields are mapped correctly.

Once you are done mapping, any unmapped fields will appear at the bottom of the page for you to check. If there is no DwC term to map these terms to, that is okay, but the data will not be published alongside the rest of your dataset. Consider moving these unmapped fields to either [dynamicProperties](#) or to one of the extensions (e.g., eMoF), whichever is most applicable. Such data would still be available if the dataset was downloaded directly from the IPT.

Finally, click Save. You can return to the Overview page by clicking Back. To add DwC mappings for the other files (Occurrence and eMoF), click the same "+ Add" button and go through the same process for each extension table you have.

The IPT may identify Redundant terms if certain terms appear in the e.g., Event core and Occurrence extension. If your Occurrence extension (or core) contains information about [individualCount](#) and [organismQuantity](#), you can map such fields in both the Occurrence and the eMoF as a [measurementType](#).

Let's watch a video of the publishing process so far.

Watch this video for a walkthrough of all steps to upload source data to an IPT, and how to map terms to Darwin Core (available <https://youtu.be/i2P8mjo128o>).

08 Publish to OBIS part 1 - Uploading & mapping to Darwin Core on a...



Play Video

In this lesson we learned what an IPT is, how to use it to upload data tables, and map our fields to Darwin Core.

Next we will learn how to fill metadata information so that our dataset can be understood by other users.

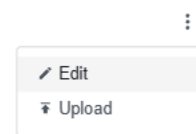
Metadata on the IPT

Metadata enables users to discover, assess, understand, and attribute your dataset for their particular needs, so it is beneficial to invest some time providing accurate and complete information here. Think about what kind of information you would want to know about a dataset if you were looking at one, or looking for datasets that fulfill a particular criteria, then apply the same logic to provide such information for your dataset. The more complete the metadata description, the better your dataset will be understood by multiple users.

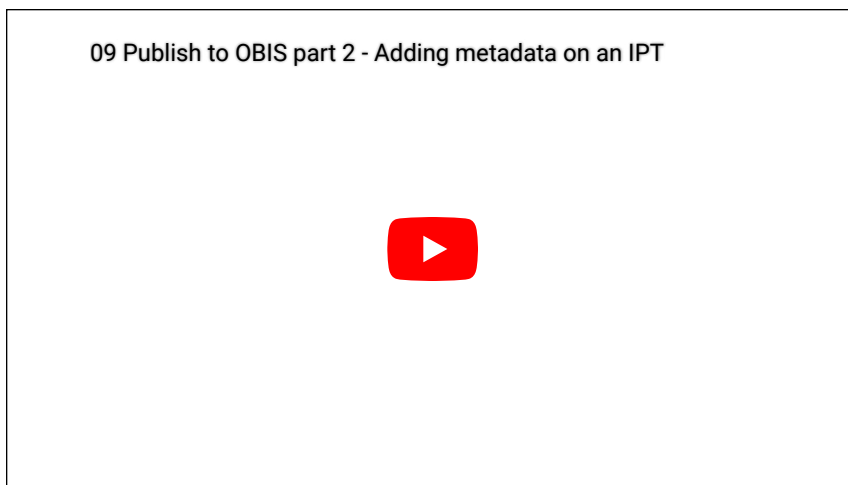
To populate metadata fields, navigate to the IPT resource overview page on the IPT, then to Metadata and click Edit to open the metadata editor. Any information you provide here will be visible on the resource homepage and bundled together with your data when you publish.

Metadata

Incomplete Your resource metadata.



Let's start by watching a video showing how to navigate and populate metadata fields. The video below (available <https://youtu.be/oAoQsDkZpS8>) provides a demonstration and discussion for filling in metadata on the IPT:



Play Video

We mentioned earlier in this course that OBIS adheres to the Ecological Metadata Language, let's learn about this standard and how it applies to your metadata next.

Ecological Metadata Language

As we have learned, OBIS (and GBIF) uses the [Ecological Metadata Language \(EML\)](#) as its metadata standard. This standard was developed specifically for the earth, environmental, and ecological sciences. EML is implemented as [XML \(eXtensible Markup Language\)](#) and is based on prior work done by the Ecological Society of America and associated efforts. OBIS specifically uses the [GBIF EML profile \(version 1.1\)](#).

To populate metadata on an IPT, you can either fill in the fields directly through the metadata editor, as demonstrated in the previous video, or you can write and upload a script in EML using R or python (e.g., using the R [eml](#) package). Regardless of which option you choose, it is good practice for your metadata to cover the following types of information:

- Geographic coverage
- Taxonomic coverage
- Temporal coverage
- Methodologies

For OBIS, the following **4 terms are the bare minimum required**: **Title**, **Citation**, **Contact** and **Abstract**. In addition to the above, there are several other categories for metadata you can provide, which includes basic information about the:

- Dataset and data provider
- Keywords
- Hosting institution information
- Information regarding associated project(s)
- Sampling methods
- How to cite the dataset
- Museum collection (if applicable)
- Other external links (e.g. a homepage) or additional metadata

Next let's look at some specific EML terms.

EML terms

Below is an overview of all the EML terms used to describe datasets, as well as the associated metadata category that is found on the IPT. You may use this table as a reference. Keep in mind the IPT usually has pop up definitions for the various terms during metadata documentation.

EML term	IPT metadata category(s)	Description
<code>title [xml:lang="..."]</code>	Basic Metadata	A good descriptive <code>title</code> to provide the user with valuable information, making data discovery easier. Multiple titles may be provided, particularly when trying to express the title in more than one language (use the "xml:lang" attribute to indicate the language if not English/en).
<code>creator ; metadataProvider ; associatedParty ; contact</code>	Basic Metadata; Associated Parties	These are the people and organizations responsible for the dataset resource, either as the creator, the metadata provider, contact person or any other association. The following details can be provided: <ul style="list-style-type: none"> • <code>individualName</code> <ul style="list-style-type: none"> ◦ <code>givenName</code> ◦ <code>surName</code> • <code>organizationName</code>: Name of the institution. • <code>positionName</code>: to be used as alternative to persons names (leave <code>individualName</code> blank and use <code>positionName</code> instead e.g. data manager). • <code>address</code> <ul style="list-style-type: none"> ◦ <code>deliveryPoint</code> ◦ <code>city</code> ◦ <code>administrativeArea</code> ◦ <code>postalCode</code> ◦ <code>country</code> • <code>phone</code> • <code>electronicMailAddress</code> • <code>onlineUrl</code>: personal website • <code>role</code>: used with <code>associatedParty</code> to indicate the role of the associated person or organization. • <code>userID</code>: e.g. ORCID. <ul style="list-style-type: none"> ◦ <code>directory</code>
<code>pubDate</code>		The date that the resource was published. Use ISO 8601.
<code>language</code>	Basic Metadata	The language in which the resource (not the metadata document) is written. Use ISO language code.
<code>abstract</code>	Basic Metadata	Brief description of the data resource.
<code>keywordSet</code>	Keywords	Includes the following: <ul style="list-style-type: none"> • <code>keyword</code>: Note only one keyword per keyword field is allowed. • <code>keywordThesaurus</code> : e.g. ASFA
<code>additionalInfo</code>		OBIS checks this EML field for harvesting. It should contain <i>marine, harvested by iOBIS</i> .
<code>coverage</code>		Includes geographic, temporal, and taxonomic coverage

	Geographic Coverage	<ul style="list-style-type: none"> • geographicDescription: a short text description of the area. E.g. the river mouth of the Scheldt Estuary. • boundingCoordinates <ul style="list-style-type: none"> ◦ westBoundingCoordinate ◦ eastBoundingCoordinate ◦ northBoundingCoordinate ◦ southBoundingCoordinate
	Temporal Coverage	<ul style="list-style-type: none"> • Use ISO 8601 <ul style="list-style-type: none"> ◦ singleDateTime ◦ rangeOfDates <ul style="list-style-type: none"> ▪ beginDate <ul style="list-style-type: none"> ▪ calendarDate ▪ endDate <ul style="list-style-type: none"> ▪ calendarDate
	Taxonomic Coverage	<ul style="list-style-type: none"> • Taxonomic information about the dataset, can include a species list. <ul style="list-style-type: none"> ◦ generalTaxonomicCoverage ◦ taxonomicClassification <ul style="list-style-type: none"> ▪ taxonRankName ▪ taxonRankValue ▪ commonName
intellectualRights	Basic Metadata	Statement about IPR, Copyright or various Property Rights. Also read the guidelines on the sharing and use of data in OBIS
purpose	Additional Metadata	A description of the purpose of this dataset
methods	Sampling Methods	<p>Includes:</p> <ul style="list-style-type: none"> • methodStep: Descriptions of procedures, relevant literature, software, instrumentation, source data, and any quality control measures taken • sampling: Description of sampling procedures including geographic, temporal and taxonomic coverage of the study. • studyExtent: Description of the specific sampling area, the sampling frequency (temporal boundaries, frequency of occurrence), and groups of living organisms sampled (taxonomic coverage) • samplingDescription: Description of sampling procedures, similar to the one in the methods section of a journal article • qualityControl: Description of actions taken to either control/ assess data quality resulting from the associated method step
project	Project Data	<ul style="list-style-type: none"> • title • identifier • personnel: The personnel field is used to document people involved in a research project by providing contact information and their role in the project • description • funding: The funding field is used to provide information about funding sources for the project such as: grant and contract numbers; names and addresses of funding sources. • studyAreaDescription • designDescription: The description of research design

maintenance	Additional Metadata	<ul style="list-style-type: none"> • description • maintenanceUpdateFrequency
additionalMetadata	Basic Metadata, Citations, Collection Data	<p>metadata</p> <ul style="list-style-type: none"> • dateStamp: The dateTime the metadata document was created or modified (ISO 8601) • metadataLanguage: The language in which the metadata document is written (<i>not</i> the resource being described) • hierarchyLevel <ul style="list-style-type: none"> ◦ citation: A single citation for use when citing the dataset. The IPT can also auto-generate a citation based on the metadata (people, title, organization, onlineURL, DOI etc). ◦ bibliography: A list of citations that form a bibliography on literature related / used in the dataset ◦ resourceLogoUr1: URL of the logo associated with a dataset. ◦ parentCollectionIdentifier ◦ collectionIdentifier ◦ formationPeriod: Text description of the time period during which the collection was assembled. E.g., "Victorian", or "1922 - 1932", or "c. 1750". ◦ livingTimePeriod: Time period during which biological material was alive (for palaeontological collections). ◦ specimenPreservationMethod ◦ physical <ul style="list-style-type: none"> ▪ objectName ▪ characterEncoding ▪ dataFormat <ul style="list-style-type: none"> ▪ externallyDefinedFormat ▪ formatName ◦ distribution: URL links <ul style="list-style-type: none"> ▪ online <ul style="list-style-type: none"> ▪ url function="download" ▪ url function="information"
alternateIdentifier		A Universally Unique Identifier (UUID) for the EML document and not for the dataset. This term is optional.

In this lesson we learned about Ecological Metadata Language, and how to fill metadata information on the IPT. Now that your dataset is uploaded, properly mapped to DwC, and all the metadata is filled in, the data can be published and made public on OBIS! We'll look at that in the next lesson.

Publish on the IPT

To publish a dataset, make sure you have navigated to the resource overview page, then go to the Publication section, click the vertical dots, and select Publish.

Publication

A preview of your pending published version compared with the current version if existing.

 Publish

Version 1.0

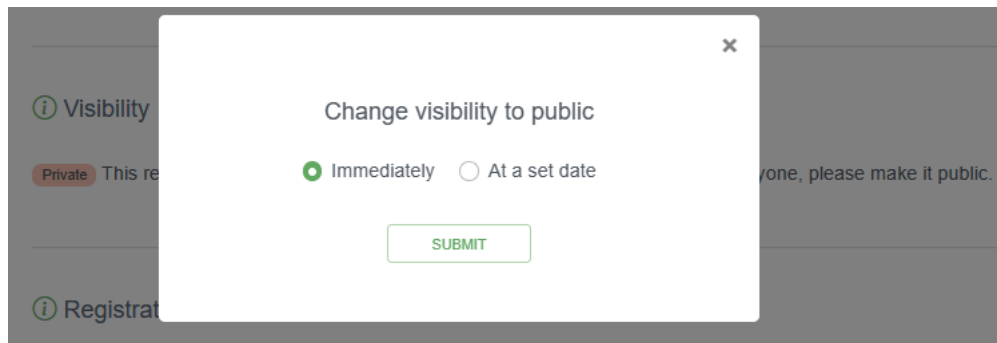
Pending - Private

Next publication date not set

The IPT will now combine the data with the metadata to package it as a standardized zip-file called a “Darwin Core Archive”.

Note: Hitting the “publish” button does not mean that your dataset is available to everyone! It is still private, with access limited to the resource managers. It will only be publicly available when you have changed Visibility to Public. You can choose to do this immediately or on a set date.

Back on the Resource Overview page > Published Release, you can see the details of your first published dataset, including the publication date and the version number. Since your dataset is published privately, the only thing left to do is to click Visibility to Public to make it available to everyone. **Warning: please do not do this with a test dataset.**



It is now listed on the IPT homepage and you can share and link to it, e.g. <http://ipt.vliz.be/resource.do?r=kielbay70>. This is a good time to notify any regional or thematic network you are involved in, which may be interested in your dataset.

Note about dataset versioning

Your published dataset is a **static** snapshot of your data and **will not change** until you upload an updated source file and click publish again or publish a new version (**do not create a new resource**). This procedure has the advantage that your dataset is always available, does not require a live internet connection to your database and can be easily shared. It also allows you to control the publication process more precisely: version 1, version 2, etc. and users are informed of how recent the data are via the last publication date. We'll discuss maintenance of datasets in the next lesson.

To view an older version of the metadata about the resource on the IPT, just add the trailing parameter `&v=n` to the URL where `v` stands for "version", and `n` gets replaced by the version number, e.g., http://ipt.vliz.be/ilvo/resource.do?r=zoo1_bpns&v=1. In this way, specific versions of a resource's EML, RTF, and DwC-A files can be retrieved. Please note, the IPT's Archival Mode must be turned on in order for old versions of DwC-A to be stored (see [Configure IPT settings](#) section of the IPT manual).

For an overview of how to publish on the IPT, watch the first minute of the video below (available at <https://youtu.be/HciufRG9hiI>). This video also contains information on how to publish to GBIF simultaneously, a topic that we will discuss in the next lesson.

10 Publish to OBIS part 3 - Publishing a dataset to OBIS and GBIF



Play Video

Data Licenses

Finally, when publishing you must choose a data license for any data made available to OBIS under one of the following Creative Commons licenses (in order of preference):

- [CC0](#) - data may be used without restrictions
- [CC-BY](#) - data are available for any use if proper attribution and credit is given
- [CC-BY-NC](#) - data may be used for any non-commercial use as long as proper attribution/credit is given

You may need to consult with your organization if there are any copyright concerns. For more information on the different Creative Commons license types see [About the licenses](#).

Maintaining resources on the IPT

Once your dataset is published and made public, you can share it and update it as often as you need to. We'll first go over how to assign a DOI to your dataset to make sharing easier, then discuss other ways of sharing and maintenance.

Adding a DOI to datasets

DOIs are important for digitally tracking your dataset. Fortunately you can easily reserve a DOI for your dataset if the IPT administrator has configured the IPT accordingly. The IPT administrator must enable the capacity for users to reserve DOIs. To do this they first need a [DataCite account](#) associated with an Organization. Only one DataCite account can be used to register DOIs in this manner (i.e. IPT users do *not* need an account). The IPT's archival mode, configurable on the IPT settings page, must also be turned on to enable this feature (note that enabling this mode will use more disk space). For more information see the [IPT administration manual](#).

Once this has been configured, you can easily reserve a DOI for a dataset by navigating to the Manage Resources tab, then select the dataset for which you wish to reserve a DOI. On the overview page for the dataset, scroll to the Publication section, click the three vertical dots and select "Reserve DOI".

Publication

A preview of your pending published version compared with the current version if existing.



Let's next review how to update your dataset with a new version.

Updating datasets

If, after publishing, your dataset has changed - extra information was added, new fields were mapped to Darwin Core, you found a more representative `measurementTypeID` code - whatever the reason, you would like to update your file. How do we do this? Fortunately, the process is largely the same as publishing your first version which we learned how to do in the previous lesson. Follow the steps below:

1. Log in to the IPT where your dataset is hosted
2. Under the Manage Resources tab, locate your dataset
3. Upload new source files to overwrite the previous ones
4. Complete the Darwin Core mapping, and update any metadata that may have changed
5. In the Publication section, click Publish

The new version will be automatically updated! The versioning may change from e.g. v1 to v1.2, or v2, etc. depending on if major changes were made, or only minor ones. There will be a pop up requesting you to confirm and describe the updates:



Are you sure?

You are about to publish a new **public minor** version of this resource. If there have been scientifically significant changes to the resource since the last publication, you should cancel, assign this resource a new DOI, and then publish again.

Assist users by summarizing what has changed in this version. You can always edit this later.

Cancel

Publish

Note: A new DOI will be generated **only if you generate it yourself** on the new version. Deciding when to generate a new DOI is up to you, but generally you should generate a new DOI when there have been major changes to your dataset, such as significant changes in your metadata or a move to a 2.0 resource version within the IPT.

Publication ⋮

A preview of your pending published version compared with the current version if existing.

Version 3.9	Version 3.10
Current CC-BY 4.0 10.25607/k68d5v Public ⋮	Pending CC-BY 4.0 10.25607/k68d5v Public ⋮
Published on 12 Mar 2023, 21:58:51	Published on 11 Apr 2023, 21:58:51

What if you want to make your dataset accessible to OBIS *and* GBIF? Let's learn how to do that on the next page.

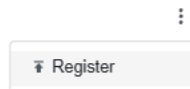
Publishing to both OBIS and GBIF

Your dataset will only be harvested and published to GBIF when you **change the Registration to Registered**. Keep in mind this step is not needed for OBIS to harvest your datasets. However, please do **not** register your dataset with GBIF if your dataset is already published in GBIF by another publisher.

Another thing to keep in mind is that **the IPT itself must be registered with GBIF** in order to publish to GBIF. The node manager can do this (see OBIS manual for IPT admin guidelines).

Registration

GBIF registration data.



Differences between OBIS and GBIF

There are a few differences between OBIS and GBIF you should be aware of. Perhaps the most obvious difference is that OBIS focuses on marine data, whereas GBIF includes broader biodiversity data. However, there are differences between quality control requirements for published datasets. A complete list of GBIF's quality control checks can be found [here](#), and a guide on GBIF's publishing process is [here](#).

OBIS currently accepts two core data table types: Occurrence core and Event core. GBIF includes both [Occurrence](#) and [Event](#) core, with one additional core type, [Checklists](#). GBIF is also developing a [new data model](#) to expand data publishing capabilities.

Some of the other main differences in how OBIS and GBIF structure and publish datasets to be aware of include:

- OBIS uses [WoRMS](#) as the exclusive taxonomic backbone, whereas GBIF uses [Catalog of Life](#)
- The OBIS-ENV-DATA structure, the eMoF extension, and the DNA Derived data extensions are not currently included in GBIF downloads (e.g., [this dataset description](#)). This data can still be published alongside your dataset, and is available when it is downloaded from the Source archive, but it will not be included in a GBIF Annotated Archive download
- OBIS conducts some QC procedures that GBIF does not, including:
 - Checking validity of depth measurements
 - Checking validity of WoRMS LSID
 - Identifying if taxa are exclusively freshwater or terrestrial
- GBIF includes most of the same [data standards](#) as OBIS (Darwin Core, EML), however GBIF also follows the [Biological Collection Access Service \(BioCASE/ABCD\)](#)

See below for a quick reference on which terms are required or recommended in OBIS and GBIF for Occurrence and Event tables.

Event Table

Term	Status in OBIS	Status in GBIF
eventID	required	required
eventDate	required	required
decimalLatitude & decimalLongitude	required	strongly recommended
samplingProtocol	strongly recommended	required
samplingSizeValue & samplingSizeUnit	strongly recommended	strongly recommended
countryCode	strongly recommended	strongly recommended
parentEventID	strongly recommended	strongly recommended
samplingEffort	strongly recommended	strongly recommended
locationID	strongly recommended	strongly recommended
coordinateUncertaintyInMeters	strongly recommended	strongly recommended
geodeticDatum	recommended	strongly recommended
footprintWKT	recommended	strongly recommended
occurrenceStatus	required in occurrence extension	strongly recommended

Occurrence table

Term	Status in OBIS	Status in GBIF
occurrenceID	required	required
eventDate	required	required

Term	Status in OBIS	Status in GBIF
scientificName	required	required
basisOfRecord	required	required
kingdom	recommended	required
decimalLatitude & decimalLongitude	required	strongly recommended
scientificNameID	required	not required, accepted
occurrenceStatus	required	not required, accepted
taxonRank	strongly recommended	strongly recommended
coordinateUncertaintyInMeters	strongly recommended	strongly recommended
individualCount, organismQuantity & organismQuantityType	strongly recommended	strongly recommended
geodeticDatum	recommended	strongly recommended
eventTime	recommended	not required, accepted
countryCode	not required, accepted	strongly recommended
informationWithheld	not required, accepted	not required, accepted
dataGeneralizations	not required, accepted	not required, accepted
country	not required, accepted	not required, accepted

This video (available <https://youtu.be/HciufRG9hiI>) will walk you through the process of publishing to GBIF as well as OBIS, how to become a GBIF publisher, and how to publish data hosted on a GBIF IPT to OBIS. It starts at 1:07.

10 Publish to OBIS part 3 - Publishing a dataset to OBIS and GBIF



Play Video

In this lesson we learned how to add DOIs to our datasets, how to update a dataset with a new file and version, and how to register our dataset with GBIF.

You have completed Module 7! To help you think about how metadata is recorded, complete Exercise 7-1

In the next module we come to the end of the OBIS data life cycle and learn how to access data from OBIS.

Module 8: Accessing OBIS data

Site: [OceanTeacher Global Academy](#)

Course: Contributing and publishing datasets to OBIS (self-paced)

Book: Module 8: Accessing OBIS data

Table of contents

Module 8

Lesson 1: Accessing data

- OBIS Mapper
- R package
- OBIS API
- OBIS Homepage
- Watch some videos
- Finding DNA sequence data
- Lesson summary

Lesson 2: Interpreting and citing OBIS data

- Contacting data providers
- Citing data
- Lesson summary

Introduction

In this module you will learn how to access and cite data from OBIS obtained from various places, including the use of the OBIS Mapper, R packages, and from the OBIS homepage.

Learning Outcomes

After successful completion of this module, you should be able to:

- Use the Mapper to download data, e.g. specific to a taxon, region, time, or type of data quality
- Use the robis R package to download data from OBIS
- Use the OBIS API to find data
- Locate a dataset in OBIS by name
- Identify data provider names and find contact information
- Construct citations for data from OBIS, construct citations for downloads of many OBIS datasets
- Understand how to navigate data downloads

How to Proceed

To succeed in this Module, you need to successfully complete the following lessons and exercises:

- Lesson 1: Accessing data
- Lesson 2: Interpreting and citing OBIS data
- [Exercise 8-1: Find OBIS data](#)

as well as successfully complete Quiz 8 with a score of $\geq 80\%$

- [Quiz 8](#)

Data Access

OBIS has over 100 million records of marine data accessible for downloading. To download data from OBIS, you have several options:

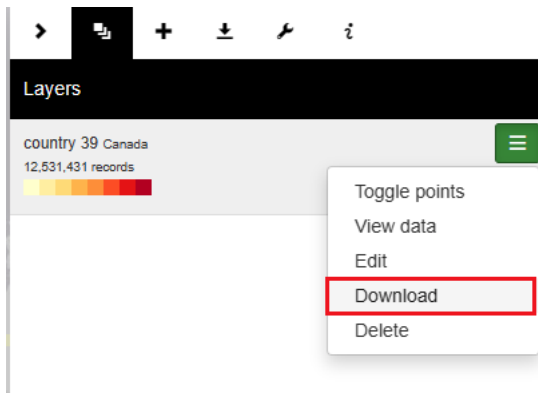
- [OBIS Mapper](#)
- [OBIS homepage](#) or [advanced dataset search](#)
- The [R package robis](#)
- OBIS [API](#)
- Full data exports
- IPT

NOTE When you download data from the Mapper or full export, the data you will receive is flattened into one table with occurrence plus event data. eMoF and DNADerivedData extension tables are separate upon request. However when you download a dataset from the OBIS homepage or dataset page, all tables (Event, Occurrence, eMoF, DNADerivedData) are included separate files.

Let's take a look at how to use the Mapper (which is different from the [Maptool](#) we learned about earlier in this course).

OBIS Mapper

The OBIS Mapper is accessible from <https://mapper.obis.org> and allows users to visualize and inspect subsets of OBIS data. A variety of filters are available (taxonomic, geographic, time, data quality) and multiple layers can be combined in a single view. You can download individual layers as CSV files, see the image below.



When you download data from the mapper, you will be given the option to include eMoF and/or DNA Derived Data extensions alongside the Event and Occurrence data. You must check the boxes for the extensions you want to include in your download.

Confirmation



Please enter your email address. You will receive an e-mail notification once your file is ready. Make sure to check your spam folder. Separate multiple e-mail addresses with a semicolon.

Select extensions to include. Note that including extensions will cause your download to take more time.

(Extended)MeasurementOrFact DNADerivedData

After downloading, you will notice that the Event and Occurrence data is flattened into one table, called "Occurrence.csv". Upon inspecting this file in your viewer of choice, you will see it contains all 225 possible DwC fields, although not every field will contain data for each observation. Any extensions you checked will be downloaded as separate data files.

Next let's learn how to use the robis package.

Data access via R package: robis

The documentation for the robis package can be found at <https://github.com/iobis/robis>. This package was developed to facilitate connecting to the OBIS API from R. The package can be installed [from CRAN](#) or [from GitHub](#) (latest development version). The package documentation includes a function reference as well as a [getting started vignette](#). As a quick example of what the package can do, you can obtain raw occurrence data by feeding a taxon name or AphiaID to the `occurrence` function.

If you'd like to then download this data, you can simply export R objects with the `write.csv` function. For example, if we wanted to obtain Mollusc data from OBIS:

```
library(robis)
```

```
moll<-occurrence("Mollusca")
```

```
write.csv(moll, "mollusca-obis.csv")
```

This file will be saved to your working directory (if you are not familiar with working directories in R, read [here](#)). After opening the file, you will notice that the fields in the download do not include every possible field, but instead only those where information has been recorded by data providers, plus the fields added by OBIS's quality control pipeline. We will learn about these fields later in this lesson.

Data access: OBIS API

Both the Mapper and the R package are based on the [OBIS API](#), which can also be accessed directly to find and download data. However we do not recommend using the API in this way to download data, it is best used for quick data summaries. When using the API directly, you can filter by the following options:

- Occurrence
- Taxon
- Checklist
- Node
- Dataset
- Institute
- Area
- Country
- Facet
- Statistics

Each of these sections have dropdowns where you can specify information to filter by. You will have to click the "Try it Out" button to enable filters. An interesting variable you can facet by is depth. This will allow you to obtain the number of records by depth bin, for all records in OBIS, a specific species, and/or a specific area. There are also a number of summary statistics you can obtain, see screenshot below.

The screenshot shows the documentation for the **GET /occurrence** endpoint. It includes a description of the endpoint, a section for parameters, and a table listing the parameters with their types and descriptions. Two input fields are shown for the parameters `scientificname` and `taxonid`.

Occurrence

GET `/occurrence` Find occurrence records.

Find occurrence records.

Parameters

Name	Description
<code>scientificname</code> string (query)	Scientific name. Leave empty to include all taxa.
<code>taxonid</code> string (query)	Taxon AphiaID.

scientificname - Scientific name. Leave empty

taxonid - Taxon AphiaID.

Statistics

GET `/statistics` Get basic statistics for occurrence records.

GET `/statistics/years` Get number of presence records per year.

GET `/statistics/env` Get number of records per SST, SSS or depth bin.

GET `/statistics/qc` Get a QC summary, including missing or invalid values, number of records on land, n

GET `/statistics/composition` Get an overview of taxonomic composition.

When you have entered all the information you are interested in filtering by, scroll down and click the “Execute” button in the same box (i.e. do not scroll to the end of the page, only to the end of the section you are applying filters to). This will produce a response detailing how many records match your criteria, as well as information for some of the headers from the data (e.g., basisOfRecord, Order, genus, etc.). A download button will be available for you to download the data as well.

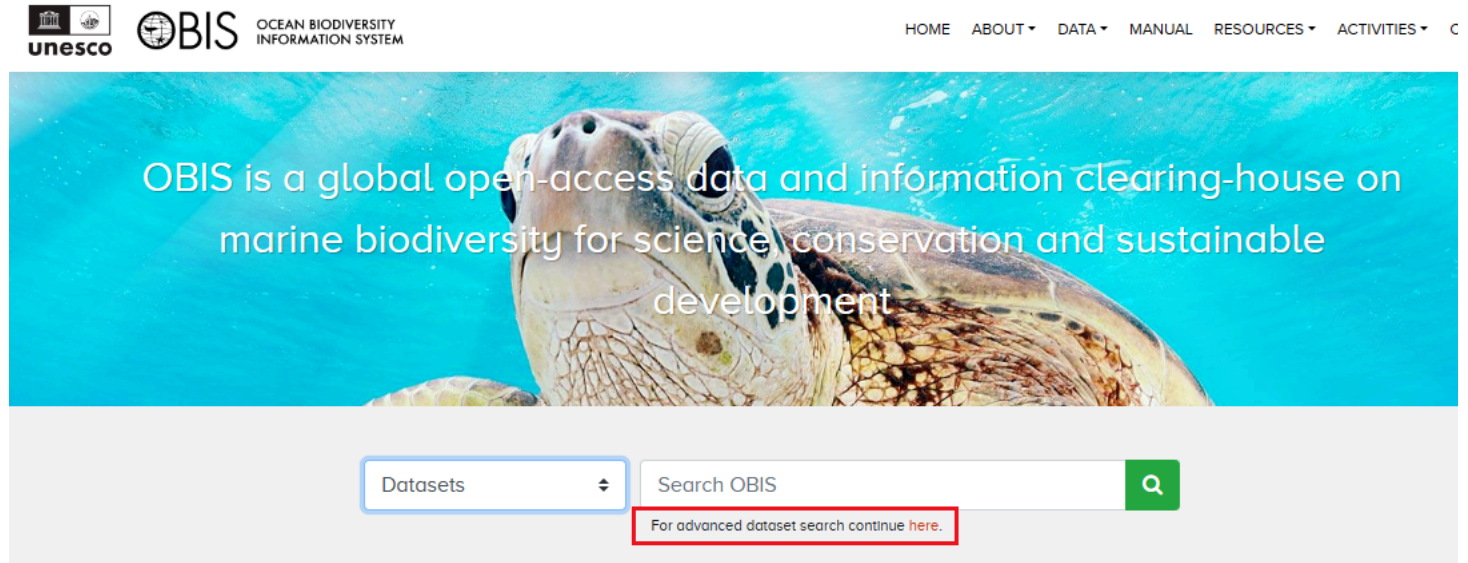
When searching with the API, you may need to know certain identifiers, including:

- AphiaID - obtainable from the WoRMS page of a taxa of interest (e.g. the AphiaID for [Mollusca](#) would be 51)
- Dataset UUID - can be obtained from the URL on individual dataset pages
 - E.g., [this dataset's](#) UUID would be 5061d21c-6161-4ea2-a8d4-38f8285dfc47
- Area ID
- Institute ID - this should be the Ocean Expert ID (e.g., the ID for [NOAA Fisheries Service, Southeast Regional Office St. Petersburg](#) is 7532)
- OBIS node UUID

Data access: OBIS homepage

From the OBIS homepage, you can search for data in the search bar in the middle of the page. You can search by particular taxonomic groups, common names, dataset names, OBIS nodes, institute name, areas (e.g., Exclusive Economic Zone (EEZ)), or by the data provider's country.

When you search by dataset you will notice an additional option appears for [advanced search options](#). This will allow you to identify specific datasets, and apply filters for OBIS nodes and whether datasets include extensions.



The screenshot shows the OBIS homepage. At the top left are the UNESCO and OBIS logos. The OBIS logo includes the text "OCEAN BIODIVERSITY INFORMATION SYSTEM". To the right is a navigation menu with links: HOME, ABOUT, DATA, MANUAL, RESOURCES, and ACTIVITIES. Below the navigation is a large banner image of a sea turtle underwater. Overlaid on the banner is the text: "OBIS is a global open-access data and information clearing-house on marine biodiversity for science, conservation and sustainable development". Below the banner is a search bar with a dropdown menu set to "Datasets", a search input field containing "Search OBIS", and a green search button. Below the search bar is a red-bordered box containing the text: "For advanced dataset search continue [here](#)."

Regardless if you found a dataset through the homepage or the advanced Dataset search, you will be able to navigate to individual dataset pages. For individual dataset pages (instead of aggregate pages for e.g., a Family) there are three buttons available:

- Report issue - allows you to report any issues with the dataset in question
- Source DwC-A - download the dataset as a Darwin Core-Archive file. This will provide all data tables as separate files within a zipped folder
- To mapper - this will open another browser with the data shown in the Mapper

New Zealand research tagging database

URL	https://nzobisipt.niwa.co.nz/resource?r=mpi_tag
Repository URL	https://nzobisipt.niwa.co.nz/
Node	SWP OBIS
Published	2018-08-08 20:59
Abstract	Tagging programmes have been used to provide information on fish and fisheries to central government policy makers in New Zealand for many years. A wide variety of species have been the subject of such studies, including finfish, shellfish and rock lobsters. In New Zealand, the Ministry for Primary Industries (formerly the Ministry of Fisheries) has funded these programmes to aid with fisheries research and stock assessment. Data from these programme are held in the "tag" database, from which the data in this dataset are sourced.
Citation	Ministry for Primary Industries (2014). New Zealand research tagging database. Southwestern Pacific OBIS, National Institute of Water and Atmospheric Research (NIWA), Wellington, New Zealand, 411926 records, Online http://nzobisipt.niwa.co.nz/resource.do?r=mpi_tag released on November 5, 2014.
Rights	This work is licensed under a Creative Commons Attribution (CC-BY) 4.0 License
Keywords	Occurrence, Observation
Contacts	Creator Kevin Mackay NIWA Contact David Fisher NIWA Metadata Provider Kevin Mackay NIWA Custodian Steward David Fisher

Darwin Core Archive as provided to OBIS I
the OBIS node

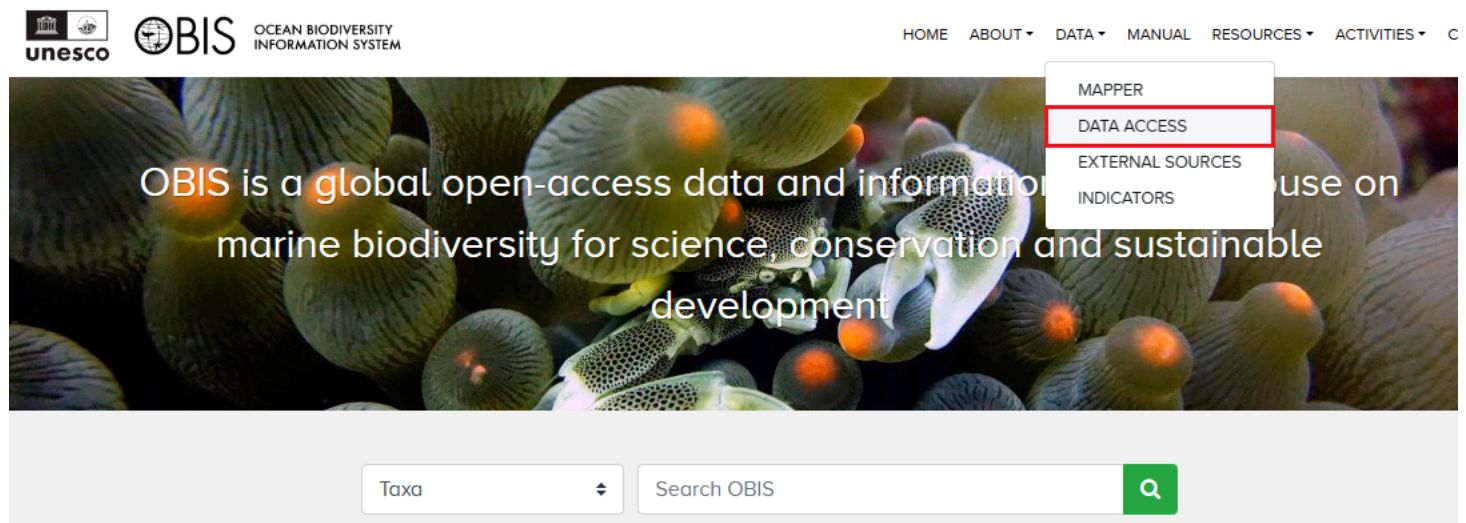
[report issue](#) [source DwC-A](#) [to map](#)

If you searched for aggregate datasets (e.g., all Crustacea records, all records from OBIS-Canada, etc.), the [source DwC-A](#) button will not be available to you. To download these data subsets, you must click [to mapper](#) and then download the data from the Mapper as a CSV as we

learned earlier.

Full exports

To obtain a full export of OBIS data, navigate to the OBIS homepage, click on Data from the top navigation bar, then select [Data Access](#) from the dropdown menu.



OBIS homepage showing where to navigate to access full database exports

Here you will be able to download all occurrence records as a CSV or Parquet file. Note the disclaimer that such exports will not include measurement data, dropped records, or absence records. As with downloads from the Mapper, the exported file is a single Occurrence table. This table includes all provided Event and Occurrence data, as well as 68 fields added by the OBIS Quality Control Pipeline, including taxonomic information obtained from WoRMS.

Watch the videos below for a demonstration on how to access OBIS data from the homepage portal and Mapper (<https://youtu.be/9PSPtqgjUI>)

11 Accessing OBIS data from the Mapper and Portal



Play Video

How to access OBIS data via R, using the obistools and Hmisc packages (<https://youtu.be/8Ep4fGICQWU>):

12 How to access OBIS data with R



Play Video

How to access OBIS data from the API (<https://youtu.be/Hocr3N6zpH0>):

13 How to use and access the OBIS API



Play Video

Accessing DNA data

As we learned earlier, you can download DNADerivedData extensions from OBIS using the Mapper. We recommend using the R package `robis` to obtain sequence data. Use the `occurrence` function and set `extensions` and/or `hasextensions` to "DNADerivedData" to ensure extension records are included in the results. `hasextensions` will exclude any occurrence that does not have the specified extension, in our case, DNADerivedData. The `extensions` parameter specifies which extensions to include. To obtain the DNA data, you have to extract the information from the extension using the `unnest_extension()` function. You can specify as many fields from the Occurrence table to be included, and pass them to the `fields` parameter. See the code below for an example. See also this [vignette](#) for a more detailed example, including how you can work further with these sequences in R.

```
dna_occ<-occurrence("Dinophyceae",hasextensions="DNADerivedData", extensions="DNADerivedData")
```

```
dnaseqs <-unnest_extension(dna_datasets, "DNADerivedData", fields = c("id", "phylum", "class", "family", "genus", "species"))
```

You also have the option to search for sequences or related sequences in OBIS by using the [OBIS Sequence Search](#). This is a prototype tool, so use caution as it is not always up to date. To use the tool, first copy your sequence in the provided box (an example sequence is provided for testing as well), like below.

Sequence

```
TAGTCATATGCTTGTCTCAAAGATAAGCCATGCATGTCTAAGTATAAGCGACTATACTGTGAACTGCGA
ATGGCTCATTAAATCAGTTATGGTTATTTGATGGTACCTTGCTACTTGGATAACCGTAGTAATCTAGA
GCTAATACATGCAGGAGTCCCGACTCACGGAGGGATGTATTATTAGATAAAGAAACCAACCGGTCTCC
GGTTGCGTGCTGAGTCATAATAACTGCTCGAATCGCACGGCTCTACGCCGGCGATGGTTTCATTCAAATTT
CTGCCCTATCAGCTTTCGATGGTAGGATAGAGGCCACCATGGCGTTAACGGGTAAACGGAGAATTAGGGT
TCGATTCCGGAGAGGGAGCCTGAGAAATGGCTACCACATCCAAGGAAGGCAGCAGGCGCGTAAATTGCC
GAATCTGCACAGGGAGGTAGTGACAAGAAATAACAATACAGGGCTATTTAGTCTTGTAATTGGAATG
AGTACAATTTACATCTCTTACGAGGATCAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGGTAATTC
```

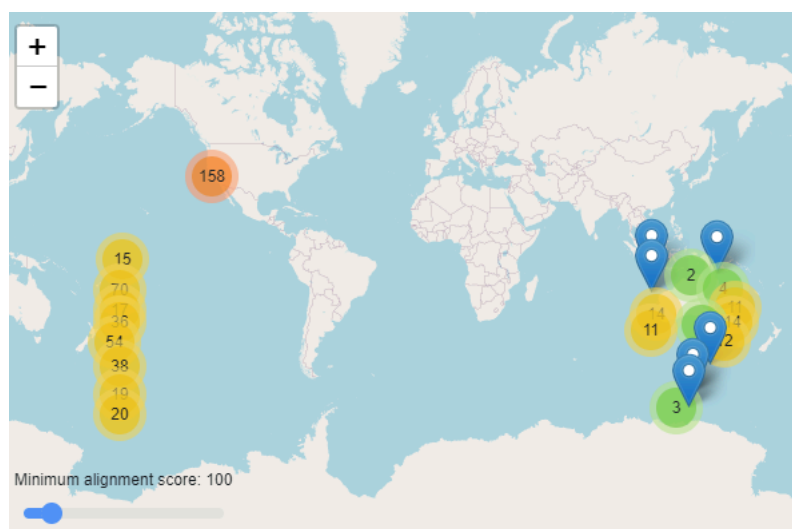
Search

About

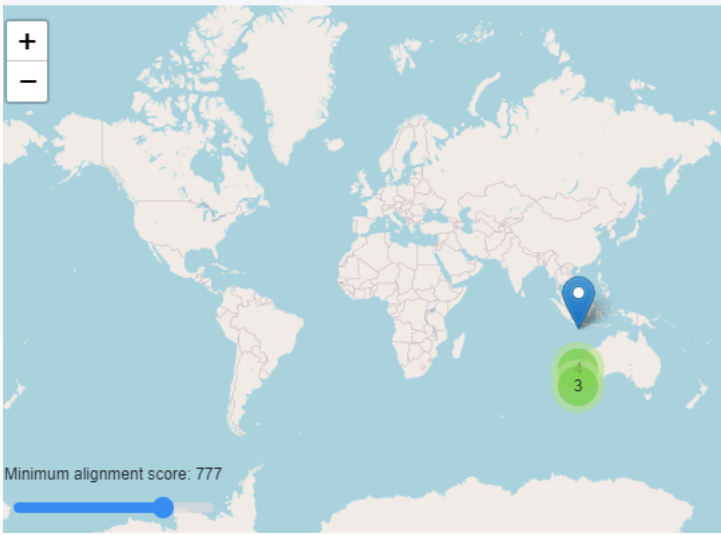
This application aligns records in the OBIS data distinct sequences as records are returned o

Then, press the Search button. Results will appear below the search box, providing you with taxonomic information on all available related sequences, in order of alignment score. The location of sequences will also be populated in the map above. Finally, a link to the associated dataset is provided in the far right column.

scientificName	alignment score	phylum	class	order	family
Prymnesiophyceae	790	Haptophyta	Prymnesiophyceae		
Algirosphaera robusta	758	Haptophyta	Prymnesiophyceae	Syracosphaerales	Rhabdosphaeraeaceae
Emiliana huxleyi	754	Haptophyta	Prymnesiophyceae	Isochrysidales	Noelaerhabdaceae
Emiliana huxleyi	746	Haptophyta	Prymnesiophyceae	Isochrysidales	Noelaerhabdaceae
Emiliana huxleyi	746	Haptophyta	Prymnesiophyceae	Isochrysidales	Noelaerhabdaceae



Note that you can also change the Minimum Alignment Score slider in the map view to adjust which sequences are shown on the map.



In this lesson we learned how to obtain data from OBIS using:

- OBIS Mapper
- OBIS homepage
- robis R package
- OBIS API
- An IPT instance
- OBIS sequence search tool

In the next lesson we will learn how to interpret and cite data obtained from OBIS.

Interpreting downloaded data

In general, the field names you will see when you download data from OBIS are the same as those seen during the data formatting and publishing process. When you download data from the [Mapper](#) you will see all 225 possible Darwin Core fields.

Downloading data from an IPT or full export will include only the fields provided by the data provider, formatted as one Occurrence file (or separate files for individual datasets). Some fields are added through the OBIS quality control pipeline, including taxonomic information from WoRMS and the fields `flags`, `bathymetry`, and `dropped`. As mentioned in the Module 4, the fields `flags` and `dropped` will list quality control issues or if the record was dropped, respectively. Some of the fields that the OBIS QC pipeline adds are shown below, but for a complete list see [here](#). Of course, for the full list of all Darwin Core terms and their definitions, please reference the [Darwin Core Quick Reference Guide](#).

field	remarks
id	Globally unique identifier assigned by OBIS.
dataset_id	Internal dataset identifier assigned by OBIS.
decimalLongitude	Parsed and validated by OBIS.
decimalLatitude	Parsed and validated by OBIS.
date_start	Unix timestamp based on <code>eventDate</code> (start).
date_mid	Unix timestamp based on <code>eventDate</code> (middle).
date_end	Unix timestamp based on <code>eventDate</code> (end).
date_year	Year based on <code>eventDate</code> .
scientificName	Valid scientific name based on the <code>scientificNameID</code> or derived by matching the provided <code>scientificName</code> with WoRMS
originalScientificName	The <code>scientificName</code> as provided.
minimumDepthInMeters	Parsed and validated by OBIS.
maximumDepthInMeters	Parsed and validated by OBIS.
coordinateUncertaintyInMeters	Parsed and validated by OBIS.
flags	Quality flags added by OBIS. The quality flags are documented here .
dropped	Record dropped by OBIS quality control?
absence	Absence record?
shoredistance	Distance from shore in meters added by OBIS quality control, based on OpenStreetMap. Negative value indicates that the observation was inland by -1 times that distance
bathymetry	Bathymetry added by OBIS. Bathymetry values based on EMODnet Bathymetry and GEBCO, see https://github.com/iobis/xylookup (Data references)
sst	Sea surface temperature added by OBIS. sst values based on Bio-Oracle, see https://github.com/iobis/xylookup (Data references)
sss	Sea surface salinity added by OBIS. sss values based on Bio-Oracle, see https://github.com/iobis/xylookup (Data references)
marine	Marine environment flag based on WoRMS.
brackish	Brackish environment flag based on WoRMS.
freshwater	Freshwater environment flag based on WoRMS.
terrestrial	Terrestrial environment flag based on WoRMS.
taxonRank	Based on WoRMS.
AphiaID	AphiaID for the valid name based on the <code>scientificNameID</code> or derived by matching the provided <code>scientificName</code> with WoRMS.
redlist_category	IUCN Red List category.

Sometimes you may want to contact a data provider for extra information regarding their dataset, let's take a look at that next.

Contacting data providers

To contact a data provider, navigate to the page for the individual dataset in question (e.g., <https://obis.org/dataset/80479e14-2730-436d-aca-a-b63bdc7dd06f>). Under the “Contacts” section, there will be a list of individuals you can contact. Clicking any name will direct you to your system’s default email program. For example:

Contacts	Creator	Todd O'Brien National Oceanic and Atmospheric Administration
	Contact	Todd O'Brien National Oceanic and Atmospheric Administration
	Metadata Provider	Abby Benson U.S. Geological Survey
	Publisher	Abby Benson U.S. Geological Survey

If you are the node manager and need to contact the data provider about a particular dataset, contact information should also be provided in the metadata.

Citing data

Depending on how you access data from OBIS, there are different ways you should cite downloaded datasets.

General OBIS citation:

OBIS (YEAR) Ocean Biodiversity Information System. Intergovernmental Oceanographic Commission of UNESCO. www.obis.org.

For **individual datasets** retrieved from OBIS (dataset citations are available in the zip downloads as html file):

[Dataset citation available from metadata] [Data provider details] [Dataset] (Available: Ocean Biodiversity Information System. Intergovernmental Oceanographic Commission of UNESCO. www.obis.org. Accessed: YYYY-MM-DD).

For example:

Sousa Pinto, I., Viera, R. (Year: if not provided use year from dataset publication date) Monitoring of the intertidal biodiversity of rocky beaches with schools in Portugal 2005-2010. CIIMAR - Interdisciplinary Centre of Marine and Environmental Research, Porto. [Dataset] (Available: Ocean Biodiversity Information System. Intergovernmental Oceanographic Commission of UNESCO. www.obis.org. Accessed: 2015-01-01)

When data represents a **subset of many datasets** taken from the integrated OBIS database (e.g. downloaded from the [Mapper](#)), you can, in addition to citing the individual datasets (and taking into account the restrictions set at each dataset level), also cite the OBIS database as follows:

OBIS (YEAR) [Data e.g. Distribution records of *Eledone cirrhosa* (Lamarck, 1798)] [Dataset] (Available: Ocean Biodiversity Information System. Intergovernmental Oceanographic Commission of UNESCO. www.obis.org. Accessed: YYYY-MM-DD)

The **derived information products** from OBIS are published under the CC-0 license and can be cited as follows:

OBIS (YEAR) [Information product e.g. Global map showing the Hulbert index in a gridded view of hexagonal cells] [Map] (Available: Ocean Biodiversity Information System. Intergovernmental Oceanographic Commission of UNESCO. www.obis.org. Accessed: YYYY-MM-DD)

In this lesson we learned how to interpret the different data fields in downloaded data. We also learned how to cite data obtained from OBIS. Practice the different ways for accessing OBIS data in Exercise 8-1 and then complete this module's quiz.

This brings us to the end of the course! To finish the course, please ensure all quizzes and exercises are successfully completed, and fill out the Feedback form so we can continue to improve this course.