JONAS
Joint Framework for Ocean Noise in the Atlantic Seas

Deliverable 4.3 - Data sharing platform

Project Name	JONAS
Project ID	INTERREG ATLANTIC - Contract EAPA 51/2018
Document name	Deliverable 4.3
Document version	1.0
Author	Ricardo Duarte and Sérgio M. Jesus
Supervisor	JONAS
Date	July 2021
Status	PU (for Public)

Interreg Atlantic Area
European Regional Development Fund
EUROPEAN UNION

Nomenclature

ADC	Analogue-to-Digital Converter
AIS	Automatic Identification System
Ax.y	Action y of work package x
CEFAS	Centre for Environmental Fisheries and Aquatic Science
Dx.y	Deliverable y of work package x
EC	European Commission
EDF	Exchange Data Format
EOV	Essential Ocean Variable
EU	European Union
GOOS	Global Ocean Observation System
GPS	Global Positioning System
HDF	Hierarchical Data Format
ICES	International Council for the Exploration of the Sea
IH	Instituto Hidrográfico
JONAS	Joint Framework for Ocean Noise in the Atlantic Seas
MSFD	Marine Strategy Framework Directive
MSS	Marine Scotland Science
OSPAR	Oslo and Paris Conventions
PC	Project Coordinator
PI	Principal Investigator
PLOCAN	Plataforma Oceanica de Canarias
PMO	Project Management Office
PSD	Power Spetral Density
QO	Quiet Oceans
SPL	Sound Pressure Level
TL	Transmission Loss
TOL	1/3-Octave Band Level
SHOM	Service Hydrographique et Oceanographique de la Marine
VRE	Virtual Research Environment
UALG	Universidade do Algarve
UCC	University College of Cork
UPC	Universitat Politécnica de Catalunya
WPx	Work Package x
WPL	Work Package Leaders

Executive Summary

The increasing awareness of the importance of sound to marine life and the understanding of the potential impact of human activities led the Oslo and Paris Commission (OSPAR) to accurately describe and frame the different issues involved in ocean acoustics and propose an important differentiation between "ocean sound" and "ocean noise". The former representing a generic term that encompasses both natural and man made sound, and the latter describes "that part of" ocean sound that is harmful or annoying for marine life.

JONAS project addresses the impact of underwater noise on sensitive species and the potential threat to biodiversity in the EU North Atlantic area. Ocean ambient sound, that stands for natural sound (physical and biological) plus anthropogenic noise, is an important instrument that allows to monitor and gather a wealth of information about the ocean itself and to assess the status of marine life. Ocean sound has now been declared by the Global Ocean Observation System (GOOS) as an essential ocean variable (EOV).

There is a broad consensus that anthropogenic ocean noise (i.e. noise generated by human activities) has increased over the past 50 years as a result of increasing human maritime activities, including seismic exploration for oil and gas, military operations, fishing sonar, harvesting of offshore renewable energy, recreational boating and mainly due to the steadily increase of ship traffic.

Throughout the years many countries and research groups are attempting to evaluate and characterize underwater noise to determine in which way this EOV impacts biodiversity. This turned out to be a daunting task due to the ocean vastness, harsh conditions in remote locations and due to constraints posed by the oceans great depth in many places. Ocean noise distribution depends mainly on noise sources intensity and location, as well as on ocean physical conditions. Low frequency noise, particularly that of shipping, propagates to long distances, across seas and oceans with no regard for political borders. Thus, ocean noise monitoring requires international cooperation between institutions and research groups, in particular through data and information exchange for building coherent sound distribution maps over large geographic areas and across extended periods of time. It is now widely accepted in the scientific community that sound maps will be effectively produced from numerical models complemented by discrete experimental data. However, open data exchange has been prevented due to the sensitivity of raw acoustic data in general and for military strategic areas in particular.

As an alternative, the JONAS project proposed that data sharing should only involve averaged data, which is sufficient for sound map calculations, and has a much lower sensitivity than raw data. To that end, there are two essential aspects to be taken into account in order to be able to mix data from different institutions and areas into extended sound maps: one is a common data calibration standard; and the other is the necessity of a full description of the data being shared (metadata), albeit not compromising data privacy.

Calibration is essential to ensure a valid comparison between sound pressure level (SPL) output, since different institutions use a variety of recording equipment with distinct electrical and mechanical characteristics. Metadata encompasses time and space coordinates of recorders, basic characteristics and conditions of performing recordings, which is essential for the understanding of the data being shared.

The identification of these shortcomings led to the development of a sharing platform under activity A4.3 of JONAS, composed of a set of procedures and the associated open source tools for promoting and facilitating data sharing between institutions. Consequently, contributing to a better understanding and monitoring of the ocean, which is reflected as the principal objective of this action.

Hence, this report describes the development of a data sharing platform by: a) defining standard algorithms (based on the algorithms proposed in PAMGuide) to transform raw acoustic data into SPL and their respective statistical quantities; b) the definition of the respective metadata; c) the definition of an high level data organization streamed into an open source low level file format that allows to group the shareable data and its metadata in one single binary file; and d) the possibility to also exchange sound map data from numerical modelling and its respective metadata.

The resulting packages that convert both raw acoustic data and model generated data into a shareable data format, named Exchange Data Format (EDF), is available for download at: www.siplab.fct.ualg.pt.

Abstract

During the last decade, underwater noise was considered as an important form of ocean pollution. This topic got under the political spotlight, leading to a greater awareness of the scientific community and therefore to an increase of acoustic monitoring efforts worldwide.

This fact highlighted the importance of cooperation between countries and institutions, particularly in what concerns data exchange. However, exchanging data may be extremely sensitive, and requires various considerations to ensure the complete description of the data being shared, its privacy, and the possibility to compare results between institutions and countries. The JONAS project developed an effort to describe underwater acoustic guidelines and standards for monitoring anthropogenic ocean noise, which will allow evaluating its impact on marine species and ultimately on ocean biodiversity.

Hence, this report describes the development of a data sharing platform combining a set of standard procedures and open source tools to facilitate data sharing between institutions. The first step encompasses the definition standard algorithms to transform raw acoustic data into averaged sound pressure level data and their respective statistics, which was implemented based on the already existent PAMGuide tool. The second step covered the definition of the metadata covering the relevant information about the data being shared, having in mind its posterior usage. The third step was to define a high level file structure to store sound maps, both from real and modeled data, in an open source shareable single standardized file.

In order to promote the wider usage and exchange of acoustic data among the underwater community, JONAS proposes also two packages based on Matlab and Python language, both composed of an interactive menu-based application and low-level standalone routines, that may be called from existing user specific code.

Contents

1 Introduction

Ambient sound is an ocean resident acoustic field, hence an important instrument to diagnose and monitor the ocean environment. The early work in ocean ambient noise may be summarized by that of Knudsen [1] (1948), Urick and Price [2] (1954), culminating with the classic paper of Wenz [3] in 1962. In these early years, difficulties were mostly associated with appropriate equipment for ocean noise measurement, requiring both long term deployments in remote areas and very low noise apparatus.

Wenz ambient noise curves, showing noise power level as a function of frequency under various ocean conditions, was a landmark. A remarkable attempt was made in order to include time and space variability by providing levels for various sea states, wind noise, rain, ice and shipping conditions. During the cold war most of the ocean ambient noise work was classified and some accounts were published only in 2005 [4]. During this period, the extreme sensitivity of underwater acoustic information for defence applications explains the low degree of data exchange between institutions and *a fortiori* between nations.

It is now well established that the increase of human activities in the ocean, such as commercial shipping, seismic exploration and sonar usage, for example, have a harmful impact on marine species and on ocean biodiversity as a whole. This impact may be linked to the influence of anthropogenic noise produced by human activities, since many marine species rely on sound to forage, to interact as a community, to mate, to orientate themselves and perceive their surroundings [5, 6].

In the past the concept of sound and noise was sometimes ambiguous and used interchangeably generating some confusion. Since 2020 OSPAR proposed to define "ocean sound" as a generic term that encompasses both information bearing signal, in terms of relevant information, and ambient noise while "ocean noise" is reserved to that component of ocean sound that is harmful or annoying for marine life [7]. This new terminology for ocean sound and noise, focus on marine life rather than on human activity.

JONAS (Joint Framework for Ocean Noise in the Atlantic Seas) is an INTERREG Atlantic Area funded project that addresses threats to biodiversity from underwater noise pollution on sensitive species in the northeast Atlantic by streamlining ocean noise monitoring and risk management on a transnational basis.

Nowadays, the efforts to promote ocean preservation and the fulfillment of associated policies, among which the Marine Strategy Framework Directive (MSFD), led to requirements for wide maritime monitoring based on sound maps drawn over large geographical areas and across extended periods of time. Clearly, this goal can only be achieved through the cooperation between institutions and nations, agreeing to share data and knowledge relative to the oceanic areas under their jurisdiction. Even at European level, the implementation of a database with ocean acoustic data never took off, despite all the efforts and incentive programs launched by the EC. There are of course various reasons for this as for example the usual bureaucracy, but more importantly, difficulties associated with data size, data description and data sensitivity.

However,the characterization of a stochastic process - such as the underwater acoustic field - is normally performed through statistical tools such as correlation functions (in time and/or space) and power spectral densities (in frequency or wave-number). Both of which require the definition of observation and averaging intervals, which defines resolution and stability of the respective estimates, and can/should be used for sound intensity characterization. Even though, in other words, sound maps do not require the exchange of raw data but only statistical indexes and averaged quantities, which allows for a substantial reduction of the amount of data to be shared and is much less sensitive than raw data. Instead it requires a common procedure for determining these statistical and averaged quantities.

Additionally, it should be emphasized that sound maps are a crude simplified representation of the intricate ocean ambient sound, without all its superposition's of man made and natural sound sources, both in the near and far field.

The work presented in this deliverable, performed under Action A4.3 - Data Sharing Platform, of the JONAS program, addresses this challenge by proposing a suite of open source sharing tools composed of: a) the adoption of an already existing common standard for data calibration (PAMGuide [8]) to account for various recording systems and algorithms for sound pressure level (SPL) calculation; b) the definition of frequency bands, time sampling/averaging intervals, spatial grids and other; c) a generic metadata format to explain the data; and d) a common open source exchange data format (EDF) for sound maps storage. The resulting software packages runs under Matlab and Python, and may be downloaded together with user manuals from this www.siplab.fct.ualg.pt.

2 Data sharing requirements

The proposed data sharing platform is based on four main aspects: a) the type of data being exchanged; b) the metadata describing the acoustic data being exchanged; c) the open access file format; and d) a set of tools to process and/or convert data into a shareable format (Fig. 1).

Projects dealing with ocean sound maps may use two different types of data, according to their origin: experimental data, obtained directly from underwater deployed recording platforms; and model generated data, resulting from extensive runs of numerical computer models. In some cases, this second data type may be obtained as a combination of experimental and modelling data, the so-called mixed data, as a result of data assimilation or calibration. While the accompanying metadata shall be in accordance with the type of data being exchanged, the format of the shareable data is completely independent of it and the only aspect that should be taken into account is the degree of detail, regarding the original data that the provider wants to share.

As a matter of privacy concern, the data sharing mechanism considered in the proposed sharing platform guarantees: a) that the owner of the data defines the data subset and the metadata to be shared with; and b) that the raw original data is protected from wide dissemination, due to sensitivity concerns or other reasons. Thus, the data owner selects a time-space-frequency data subset from the raw data and converts it to averaged sound pressure level and its statistics over time, which contains all the relevant characteristics of the ocean sound field, while precluding the possibility of access to the original raw information, combined with the respective metadata in one single file (Fig. 2).

Possibly with less privacy restrictions, the same applies to the model generated data which incorporates a substantial know-how and often requires the usage of other data, such as AIS, source level, oceanographic or geo-acoustic data sets. In this case, it is

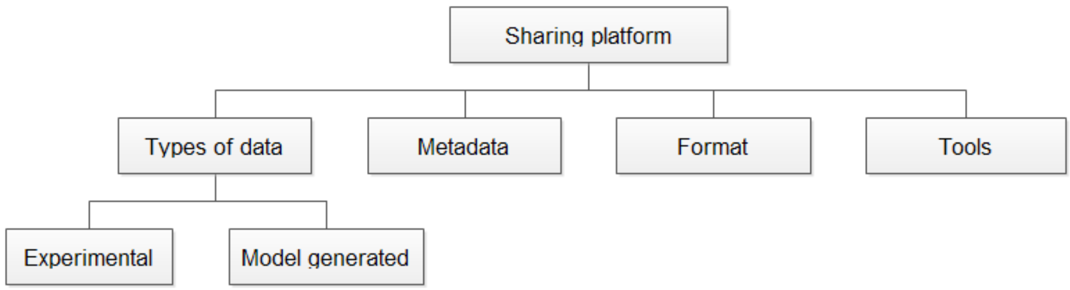


Figure 1: Diagram of sharing platform components: data, metadata, data format and processing tools. Two data types: experimental data and model generated sound maps, that include mixed data.

important to state that model generated data includes the data that is assimilated or calibrated through the usage of experimental data at selected points in time, space and frequency.

A way to respond to this commitment, and consequently allow to exchange ocean sound relevant data information between project partners, collaborators and stakeholders, is through the use of a dedicated format : the Exchange Data Format - EDF (Fig.2).

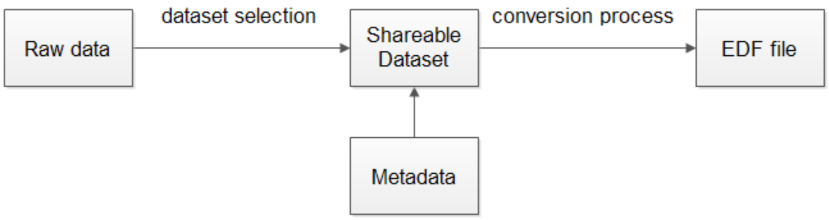


Figure 2: Processing steps from raw data to exchange data format (EDF) file.

The data sharing platform proposed under the JONAS project is a high level set of descriptors resulting on a single Exchange Data Format file, that has the following functional requirements :

1. a standard procedure for calculating sound pressure levels from calibrated raw acoustic data;
2. sharing of sound pressure levels, both from measured acoustic data and from model predictions;
3. calculating and sharing of statistical quantities from sound pressure levels;
4. defining formats for sharing additional information such as location, time and frequency to allow precise sound mapping, comparison and integration between data sets;
5. use of a low level data format that is both open source, compact and widespread in the engineering and ocean science community;
6. implementation through a set of open-source routines in open source software and or menu-driven application

3 Acoustic data types

As presented in section 2 the data sharing platform considers two independent types of data, experimental and model generated, in which its particularities will be described in the following subsections.

3.1 Experimental acoustic data

Considering experimental acoustic data, it is important to analyse the complete signal path from the acoustic sensor up to the resulting shareable data file, while giving the requirements for processing and standardization at each level. The data path can be divided in two phases as shown in Fig. 3) and which will be detailed in sub-sections 3.1.1 and 3.1.2, respectively.

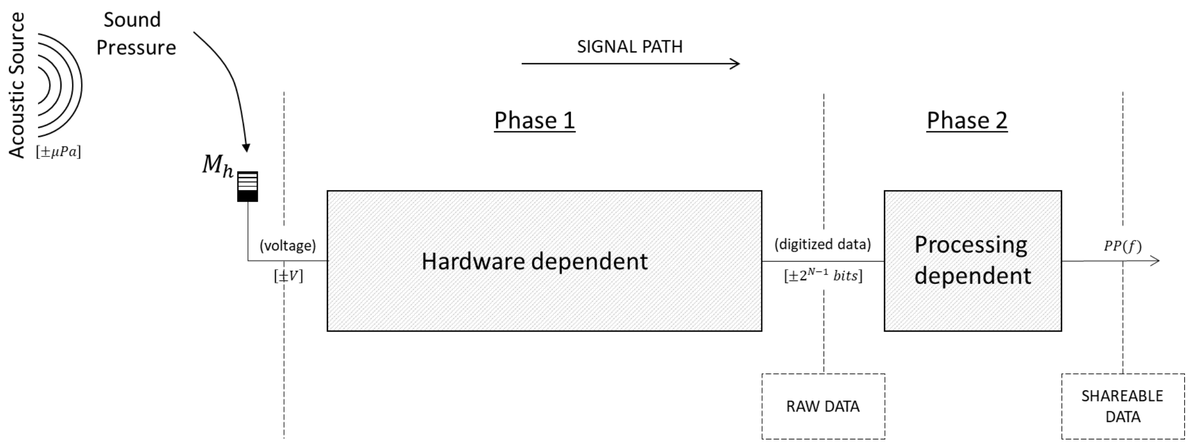


Figure 3: Data path diagram from sound pressure to shareable data: data calibration phase and raw data conversion phase.

3.1.1 From hydrophone to raw data

The first phase describes the sequence of processing stages starting with the signal received by the hydrophone up to the raw data file (see details in Fig. 4). An hydrophone is an underwater electro-acoustic transducer (the terrestrial equivalent is a microphone), that converts sound pressure of the surrounding environment into an electrical voltage. This electrical voltage may be transformed through multiple stages and components with a variety of setups according to its usage, platform installed, etc, but it may be simply described by the following steps: the signal passes through an amplifier to increase its amplitude by a specific gain, then converted into a digital stream by the analog-to-digital converter (ADC) and finally is stored in a digital support, such as a memory card, hard disk or other. In order to be able to quantify which sound pressure level corresponds to a given digital level in the audio file it is of paramount importance to perform data calibration. The detailed information regarding calibration is available in the Technical Report - Acoustic data gathering stations: section 4 - Recommendations for raw data

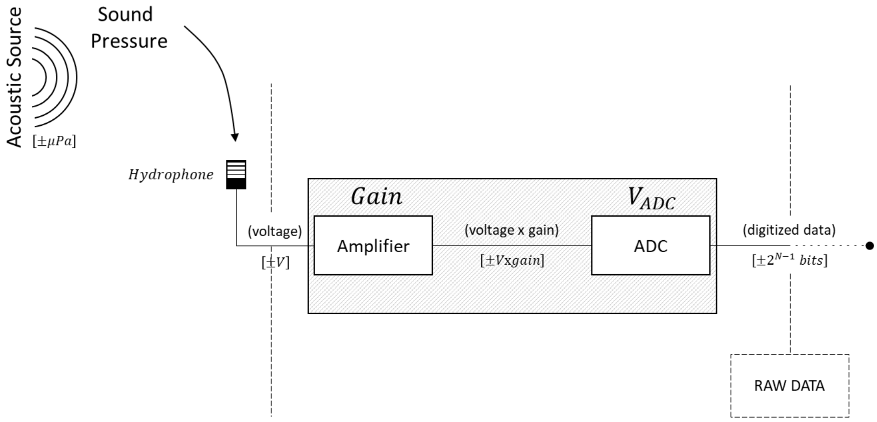


Figure 4: Phase 1 - Calibration path.

formatting and instrument calibration [9] and requires basically three values: the microphone sensitivity (in dB re 1V/μPa), the chain gain (in dB) and ADC input sensitivity voltage (in Volts peak to peak).

3.1.2 From raw data to shareable data

The second phase corresponds to the transition from raw data to shareable data and includes, firstly, the necessary processing steps to transform raw data into comparable SPL, and secondly the addition of metadata information, as a detailed description of the data being shared (Fig. 5).

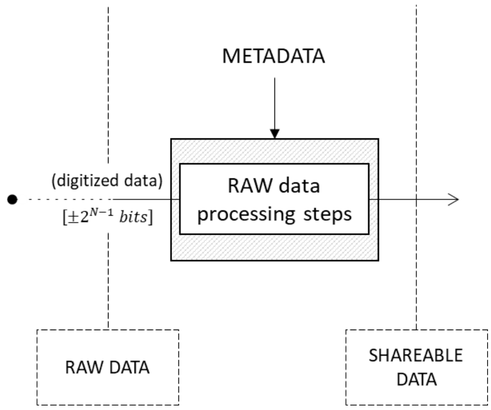


Figure 5: Phase 2 - raw data conversion into shareable data.

To be able to compare results and share the information among institutions, it is necessary to ensure that shared data is comparable and its processing gives equivalent results. For this reason the resulting shareable file must include the following quantities:

1. sound pressure level (SPL) for standard frequencies (considering 1/3 octave-band levels) and averaging times (1s), expressed in dB re 1μPa^2/Hz;

2. the same percentiles, calculated from a statistical analysis of SPL through time.

Currently there are several tools available to perform these tasks, but one that has attracted attention is PAMGuide [8]. This tool is based on open access code in MATLAB (an R version also exists) that allows to convert raw acoustic data into SPL and draw associated statistical quantities (percentiles) through the usage of standard algorithms which are described in Appendix A. Beyond data processing, transforming raw data to shareable data, needs to include data describing the data being exchanged, the so called metadata. Metadata should contain the information regarding the data acquisition process (e.g. data provider, institution, contact person), periods of time considered, frequencies covered, spatial coordinates, etc, organized in a logic way beyond all other data type dependent relevant information required for post processing.

3.2 Model-generated acoustic data

Exchanging SPL sound maps generated by numerical models is a much more direct procedure, although less transparent since, in this case, the processing is provider dependent. In fact, model-generated data is not *stricto sensu* modelled data, since most models use experimental data from live oceanographic data or historic information from databases for water column temperature and salinity, bathymetry, bottom geo-acoustic properties, and also live data from AIS. Additionally, model-generated data results result, in many cases, from assimilation process between the strict model output and experimental data (the so-called mixed data).

4 Required metadata

Metadata is a fundamental piece of additional information for the understanding of the data being shared. It should summarize the basic information about data and also provides additional information for data display, tracking and comparison. Metadata is divided in two main sections: a) generic information and b) specific metadata. While the generic metadata describes the information related with the data provider, as for example the institution, the concerned period of time, the purpose of the analysis and the internal reference of that particular dataset; specific metadata contains a more detailed description regarding the data being exchanged, as for example: the equipment used, calibration, x-y-z spatial coordinates of the area, time and frequency axes, averaging and statistical parameters, etc.

5 Shareable data format proposal

Based on the requirements above the present section defines the JONAS project shareable data format proposal named as Exchange Data Format (EDF).

5.1 Exchange Data Format fundamentals

The EDF proposal is a high level file descriptor based on HDF5 (Hierarchical Data Format version 5). HDF5 is a low level open access file format currently used for a wide range of applications whenever the manipulation of large sets with multiple data streams is required (www.hdf5.org).

The choice of HDF5 is justified by its adoption by the International Council for the Exploration of the Sea (ICES) for the registration of continuous underwater noise, while OSPAR may also join the database hosted by ICES. HDF5 follows a strict hierarchical folder structure which allows to store and organize the information in three categories (as shown in Fig. 6):

1. groups: directory tree that might contain other groups or datasets;
2. datasets: a variable that contains data;
3. metadata: a description pane that gives information about each dataset or group.

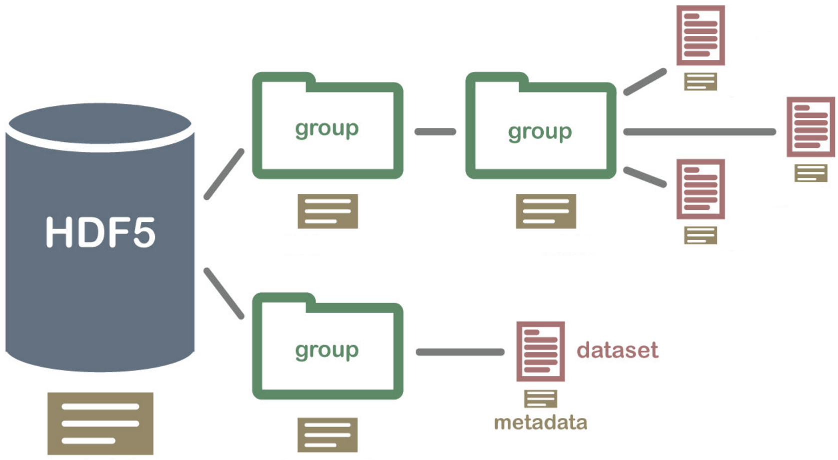


Figure 6: HDF5 file structure (adapted from www.hdf5.org)

.

5.2 Exchange Data Format proposal

Based on the low level HDF5 file structure, the high level EDF file was developed to integrate the requirements of each data type in a logic and flexible structure, giving the data owner the opportunity to easily identify the required fields. Thus, the main sets of information were defined as follows (see Fig. 7):

1. **generic information:** aspects related with the identification of the file and the data provider. This field is always present;
2. **analysis metadata:** information specific to the data type being exchanged. There are two sub-groups of parameters to be filled alternatively if the file contains experimental data or if the file contains model-generated data. In the event of mixed data (see description above) both metadata fields should be filled.
3. **ocean sound maps:** the structure of this information package is common to all types of data files and mainly contains the SPL time-frequency-spatial "cube" and associated statistical information (percentiles).

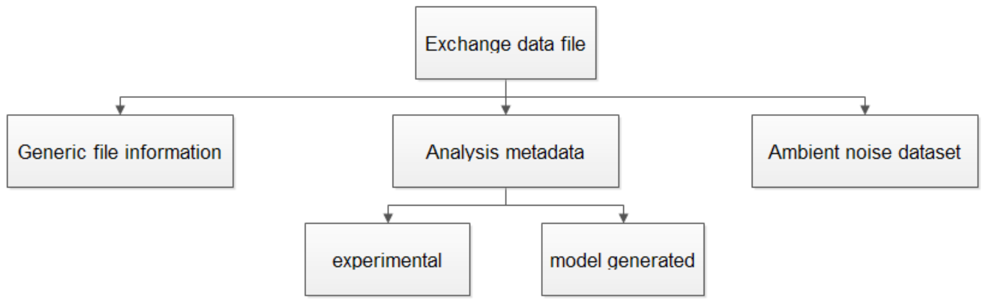


Figure 7: Exchange data file structure

The complete field description is presented in Appendix D.

6 EDF tool package

The EDF tool package was organized to support users into writing and reading exchange data files in EDF format. Two alternatives are given: a) an interactive interface and b) standalone routines (Fig. 8).

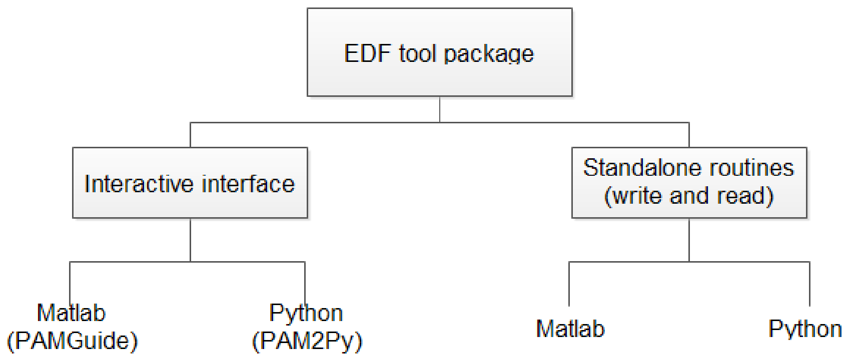


Figure 8: EDF tool package

The interactive interface was designed for occasional users that desire an intuitive and easy to use stable suite of algorithms to process raw or modeled generated data and save the output into EDF files. In the case of experienced users that already have their processing setup, the EDF tool package includes a set of standalone subroutines designed to read and write EDF files. EDF files may be read and written interchangeably by the interactive interface code or the standalone routines.

Both the interactive package and the standalone routines are available as m-files for Matlab and .py files in Python. Even though, Matlab is commonly used in the scientific community providing a suite of analysis and visualization tools, it requires a complex and lengthy installation, subject to proprietary licensing. Python is nowadays becoming an alternative for open source data analysis, that is widely supported by a live community of users.

These two options are setup as follows:

- Matlab version: PAMGuide[1] that, besides PAMGuide file saving options, also allows to write EDF files;
- Python version: PAM2Py that is a transcription of the Matlab version of PAMGuide to Python.

6.1 Interactive interface structure

The interactive interface was designed according to the following flowchart (Fig. 9). In the experimental data branch, a single or multiple (in batch mode) .WAV or .FLAC file(s) may be processed and the respective output is written in express format (.csv) without metadata or in exchange data format EDF (.h5) containing data plus metadata. On the model generated data branch, data is read from .mat files. Note that, as explained above, in this case there is no processing since the input files contain processed data already and the program fulfils the role of a format converter. The resulting file will then be written exclusively in EDF. A detailed user guide is provided in appendix B.

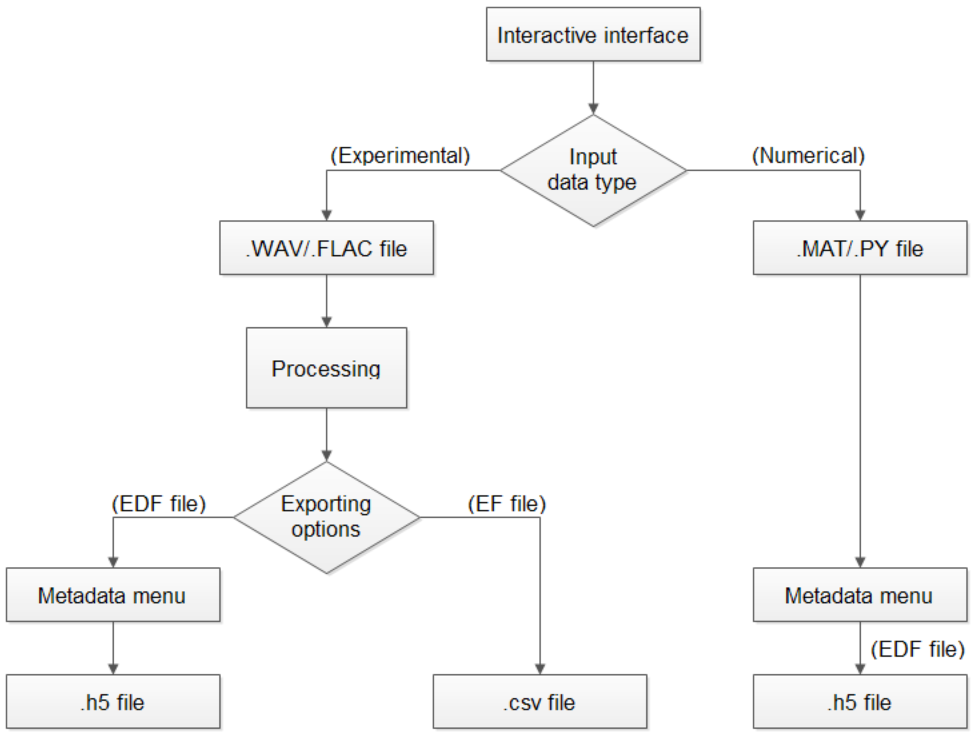


Figure 9: PAMGuide and PAM2PY usage flow diagram.

[1]Merchant, Nathan D. and Fristrup, Kurt M. and Johnson, Mark P. and Tyack, Peter L. and Witt, Matthew J. and Blondel, Philippe and Parks, Susan E., Measuring acoustic habitats, in Methods in Ecology and Evolution (2015), pp. 257–265.ISSN: 2041210X.DOI:10.1111/2041-210X.12330

6.2 Input data

6.2.1 Experimental data

The experimental input data contemplated is exclusively in non compressed .WAV or .FLAC formats. In this case its considered only raw data and for this reason it is important to define the experimental setup as well the specifications of the equipment used to record and the performed calibration. Additionally, dealing with raw data requires several processing steps which are fully described in Nathan et al. [8].

6.2.2 Model generated data

Model generated data is produced by numerical models as an attempt to forecast sound maps, often using AIS, wind, bathymetry and water column data relative to the area and time interval of interest. In this case, the input numerical file is tool dependent (.MAT only). Note that in this case the input data is already processed according to the data provider specifications.

6.3 Metadata window

The metadata window is only displayed if the EDF file exporting option is selected and once the processing steps are over. Fig. 10 shows an example of the fields considered, which are fully described in appendix B.3.

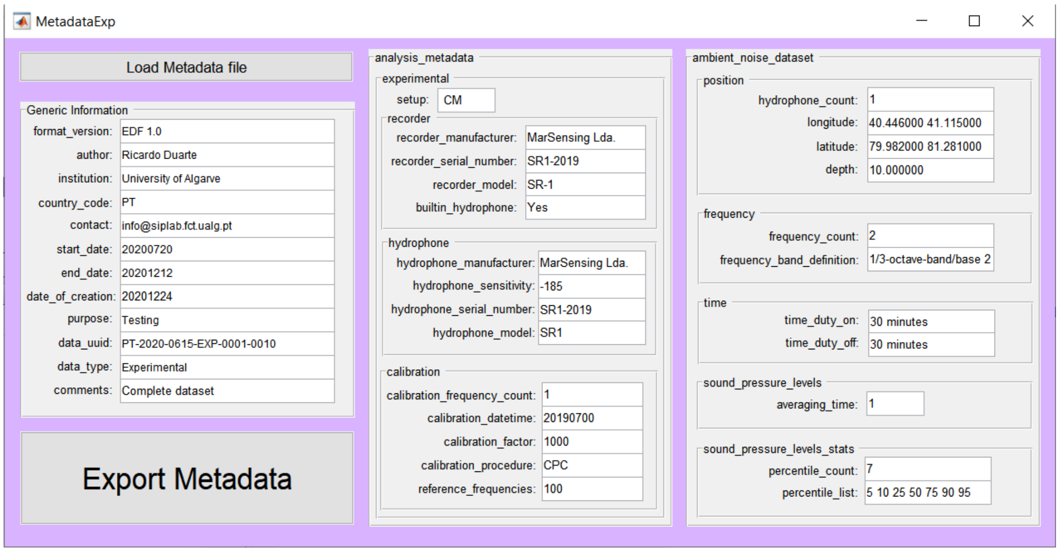
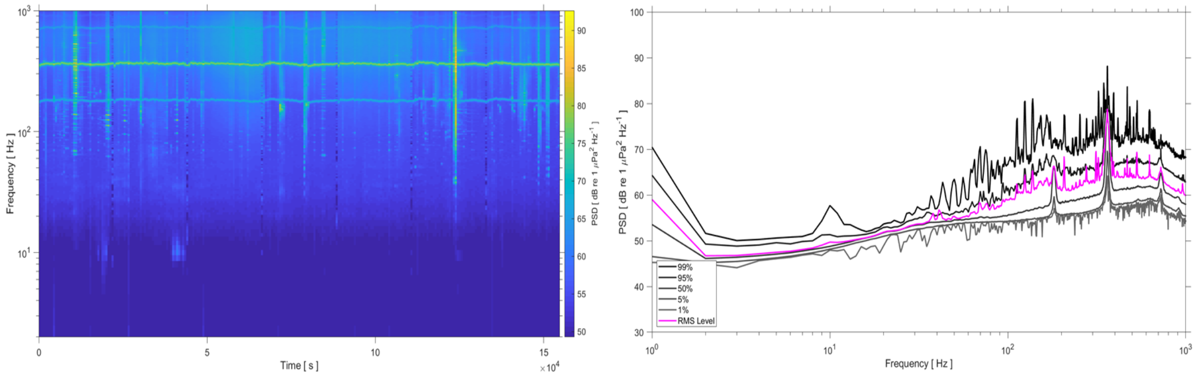


Figure 10: Metadata pop-up window.

6.4 Output results

The tool generates two types of outputs a) shareable data files and b) plots. In the case of shareable data files two outputs may be obtained depending on the exporting option selected (EDF or EF option). If EDF option was chosen users will obtain a .h5 file containing the ambient noise dataset and the complete metadata organized as shown in the Table 9. On the other hand, if EF exporting option is chosen, users will obtain a .csv file containing the SPL levels. The graphical output is divided in spectrogram and power level statistics, and is manly used as data previewer (an example of which is shown in Fig. 11b).



(a) (b)

Figure 11: EDF graphical output: spectrogram (a) and percentiles (b).

7 Conclusion

Rather than undertaking experimental activities for ocean sound recording, the JONAS project foresees the usage of experimental acoustic data already available among the different partners. However, exchanging underwater acoustic data is sensitive, which in many cases precludes its usage for scientific purposes, in particular for evaluating the impact of underwater noise on marine life. A possible way to overcome this issue is to exchange averaged noise levels in frequency bands, along time intervals and sparse spatial grids instead of raw data. Therefore, it is of paramount importance to define a strategical platform allowing for data exchange between institutions.

The work described in this report is done under action A4.3 of JONAS work program that foresees the need and sets the requirements for the proposed platform. It encompasses the definition of a proposed exchange data format (EDF) and develops a complete tool package that all together forms the data sharing platform.

This packaged features averaged noise levels, standardized statistical indicators and holds both experimental, model-generated or mixed sound maps. These various datasets are accompanied by metadata description fields to allow posterior data understanding, tracking and visualization. Due to the necessity to ensure data-piecing from various institutions the proposed EDF tool package is based in standard algorithms to convert raw acoustic data into sound pressure levels as proposed by the PAMGuide tool. Further, the EDF tool package is available both for Matlab and for Python, either as an interactive easy to use application or as standalone routines for easier integration. The complete EDF tool package and user manuals is available at: www.siplab.fct.ualg.pt.

As per the JONAS work program the work developed in this action will contribute to facilitate the integration of data into the Virtual Research Environment (VRE) platform developed in WP3.

References

[1] W. Knudsen, R. Alford, and J. Emling, "Underwater Ambient Noise.," *Journal of Marine Research*, vol. 7, pp. 410–429, 1948.

[2] R. J. Urick, "The Backscattering of Sound from a Harbor Bottom," *Journal of the Acoustical Society of America*, vol. 26, no. 2, pp. 231–235, 1954.

[3] G. . Wenz, "Acoustic Ambient Noise in the Ocean : Spectra and Sources," *The Journal of the Acoustical Society of America*, vol. 34, no. 12, pp. 1936–1956, 1962.

[4] W. Carey, "Sound sources and levels in the ocean," *Oceanic Engineering, IEEE Journal of*, vol. 31, pp. 61 – 75, 02 2006.

[5] D. P. Nowacek, F. Christiansen, L. Bejder, J. A. Goldbogen, and A. S. Friedlaender, "Studying cetacean behaviour: new technological approaches and conservation applications," *Animal Behaviour*, vol. 120, pp. 235–244, 2016.

[6] A. J. Read, "New approaches to studying the foraging ecology of small cetaceans," *Developments in Marine Biology*, vol. 4, no. C, pp. 183–191, 1995.

[7] M. van Oostveen, D. Bardé, and J. Kwakkel, "OSPAR candidate indicator ambient underwater sound," tech. rep., 2020.

[8] N. D. Merchant, K. M. Fristrup, M. P. Johnson, P. L. Tyack, M. J. Witt, P. Blondel, and S. E. Parks, "Measuring acoustic habitats," *Methods in Ecology and Evolution*, vol. 6, no. 3, pp. 257–265, 2015.

[9] R. Duarte and S. M. Jesus, "Acoustic data gathering stations," tech. rep., 2020.

[10] D. A. Abraham, *Underwater Acoustic Signal Processing: Modeling Detection, and Estimation.* 2019.

[11] S. M. Jesus, "Universidade do Algarve Acoustic pressure and particle motion power spectrum estimation," 2019.

[12] H. Li, *Spectral Analysis of Signals [Book Review]*, vol. 24. 2008.

[13] S. M. Alessio, *Digital Signal Processing and Spectral Analysis for Scientists.* 2016.

[14] ANSI, "Specification For Octave-band and Fractional-octave-band Analog And Digital Filters.," tech. rep., 2009.

[15] P. C. Etter, *Underwater Acoustic Modeling and Simulation.* 2003.

[16] K. Betke, T. Folegot, R. Matuschek, J. Pajala, L. Persson, J. Tegowski, J. Tougaard, and M. Wahlberg, "BIAS Standards for Signal Processing. Aims, Processes and Recommendations.," tech. rep., 2015.

[17] A. Van der Graaf, M. Ainslie, M. André, K. Brensing, J. Dalen, R. Dekeling, S. Robinson, M. Tasker, F. Thomsen, S. Werner, A. Van der Graaf, M. Ainslie, M. André, K. Brensing, J. Dalen, R. Dekeling, S. Robinson, M. Tasker, F. Thomsen, and S. Werner, "European Marine Strategy Framework Directive - Good Environmental Status (MSFD GES): Report of the Technical Subgroup on Underwater noise and other forms of energy.," tech. rep., 2012.

Appendices

A Processing basic notions

This section describes some basic notions regarding the signal processing steps used in the Deliverable D4.3 of the JONAS project.

A.1 Power Spectrum Estimation

To examine the amplitude vs frequency characteristics of a window function and to estimate the spectral density of a signal, the periodogram method is used [10] [11]. However, since the periodogram provides a power spectral density (PSD) with a high variance, several methods have been proposed for alternatives, such as that proposed by Daniell (1946), Bartlet (1948) and Welch (1967) which are all based on averaging the sample power spectrum over a series of subsequent or overlapped time intervals [12] [13]. The Welch method is basically obtained by refining the Bartlett method in two aspects:

- data segments are allowed to overlap by an amount of 50%;
- Hanning window is used instead of the trivial rectangular window.

The following expression was used to define the sample power spectrum (eq.1), the single-sided power spectrum (eq.2) and the Power Spectrum estimate (eq.3).

- Sample power spectrum computed from the DFT for m^{th} segment, where N is the number of samples in each segment:

$$p^m[k] = \frac{|X^m[k]^2|}{N}, \tag{1}$$

- Single-sided power spectrum (Parseval's Theorem scaling method) where $0 < k < \frac{N}{2} - 1$:

$$P_{ss}^{(m)}[k] = 2P^{(m)}[k], \tag{2}$$

- power spectrum estimate:

$$P_{ss}[k] = \frac{1}{M} \sum_{m=1}^{m=M} P_{ss}^{(m)}[k], \tag{3}$$

The sample power spectral density (PSD) (eq.4) is the distribution of power given by the sample power spectrum ($P_{ss}^{(m)}[k] = 2P^m[k]$) within each frequency interval Δf where $B = \frac{1}{N} \sum_{n=0}^{N-1} \left(\frac{w[n]}{\alpha}\right)^2$ is the normalization factor related with the window used and $\Delta f = \frac{F_s}{N}$ is the frequency bin width and is given the by:

$$PSD[k] = \frac{1}{M} \sum_{m=1}^{m=M} 2\frac{|X^m[k]|^2}{NB\Delta f}. \tag{4}$$

A.2 Frequency bands

This section describes the frequencies at which 1/3-octave band levels were assessed according to the American National Standards Institute [14].

- 1/3 octave band levels
$$f_{ci} = f_{ref} 10^{\frac{i-1}{10}}, \tag{5}$$
where f_{ref} is the standardized reference frequency of 1kHz, $i \geq 1$ corresponds to $f_c \geq 1kHz$ and $i < 1$ corresponds to $f_c < 1kHz$.

- Upper bound
$$f_{upper} = f_{ci} 10^{\frac{1}{20}}, \tag{6}$$

- Lower band
$$f_{lower} = f_{ci} 10^{\frac{-1}{20}}. \tag{7}$$

A.3 Sound pressure level

Sound pressure level (SPL) is defined as the sound intensity of a plane wave [10, 15]. Assuming a constant water impedance, sound intensity may be replaced by sound pressure, in Pascal [Pa] or microPascal [μPa] units. Considering the reference power pressure in water ($p_{ref}^2 = 1\mu Pa$) and the power spectrum estimates (P$_{ss}$), defined above, it is possible to define the SPL measured in dB re $1\mu Pa^2$ as:

$$SPL(f) = 10log_{10}\left(\frac{P_{ss}}{p_{ref}^2}\right) - S(f), \tag{8}$$

where $S(f)$ is the calibration correction factor defined in the JONAS Technical Report - Acoustic data gathering stations [9]. However, many studies regarding to the noise impact in mammals auditory system uses the concept of critical bandwidth [16]. For this reason, it is important to compute the SPL taking into account fractional-octave-band levels [17] which is the sum of the power in all adjacent 1/3-octave band levels covering the entire bandwidth and equals the total power of the signal. Thus, the SPL in 1/3-octave band levels is given by

$$SPL(m, f_c i) = 10log_{10}\left(\frac{1}{p_{ref}^2}\sum_{k=f_{lower}}^{k=f_{upper}}\frac{P_{ss}^{(m)}(k)}{B}\right) - S(f_{ci}), \tag{9}$$

where $P_{ss}^{(m)}(k)$ is the power spectrum of the m^{th} segment, f_{upper} and f_{lower} are the upper and lower bounds of each 1/3-octave band centre frequencies computed directly from the time-domain (detailed in the Appendix B) and $S(f_{ci})$ the calibration correction factor.

A.4 Percentile level

Percentile is a common statistical indicator used in underwater noise to define the level of noise exceeded during a specific percentage of time in the considered interval. JONAS project partners agreed to considered the following list of percentiles: 5, 10, 25, 50, 75, 90 and 95. Then, to obtain the percentiles it was necessary to calculate the SPL histogram over a given interval of time that is representative of the empirical probability density and then deduced the inverse distribution function between 0% and 100% which consequently returns the percentile values.

B EDF tool package manual

This section covers the basics to get started with sound field analysis using the interactive interfaces of PAMGuide (Matlab version) and PAM2Py (Phyton version). You may refer to the following usage chart in Fig. 12 for guidance. Both experimental data (single or multiple .WAV or .FLAC format) and model generated data (in .MAT file format) can be used as an input. The SPL output can be written in .h5 or .csv format, depending on the intention to share or not the accompanying metadata respectively.

Note that an important piece of information for EDF is the metadata, that may be read/saved on file or introduced through a specific menu. The metadata required for experimental or for model generated data is different so are the respective menus for input.

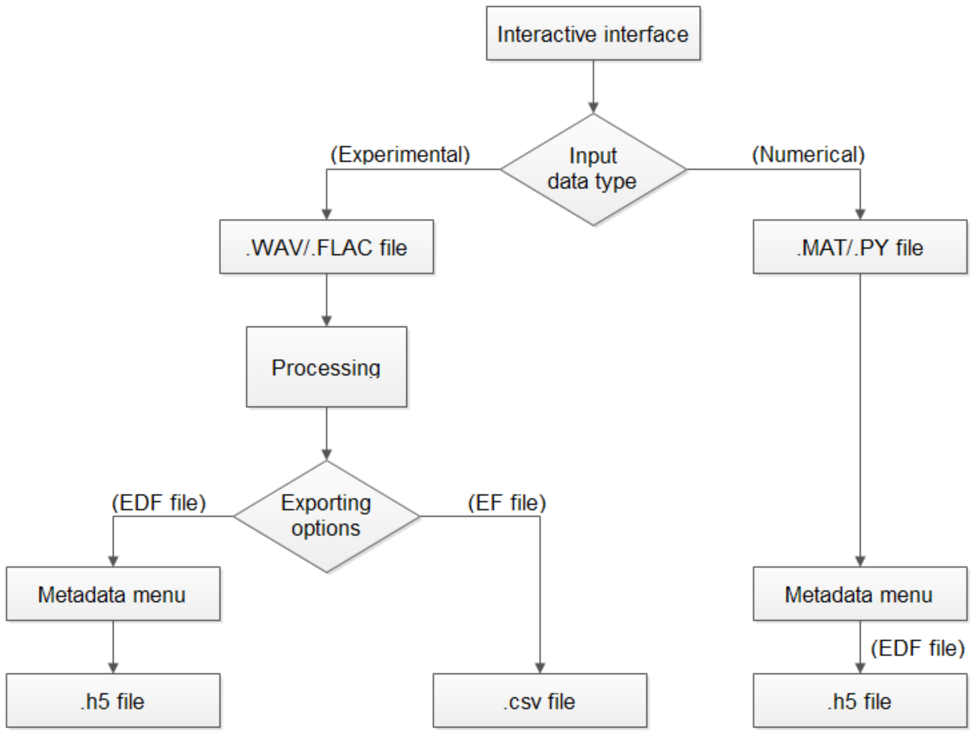


Figure 12: PAMGuide and PAM2PY usage flow diagram.

In case of working with experimental data only single or multiple WAV/FLAC files can be considered. The data will be processed and recorder calibrated exactly as in the previous version of PAMGuide in agreement with the options selected by the user in the respective menus.

If EF (i.e. Express Format) is chosen, a .csv file will be generated as an output. In this case the resulting file will be written exactly as in the standard PAMGuide. Alternatively if EDF is selected for output, metadata will be requested. A menu will open and default metadata will be read from a file with the same name as the input file but with the `.met` extension. If such a file does not exists in the current directory the user may either input data by hand directly on the provided menu, or select a metadata file for input with another name or on another directory. The user should check the metadata and change it according to the needs. When finished metadata will be saved on a file with the name of the input file and the `.met` extension. If batch processing is requested the user selects an input folder instead of a file, and PAMGuide will work through the folder and process all existing WAV/FLAC files, sequentially as they appear in the folder and produce a single EDF file as output.

In the case of working with model generated data only MAT files can be considered. Note that model generated data is that produced by numerical models as an attempt to forecast noise maps, often using AIS, wind, bathymetry and water column data relative to the area and time interval of interest. In what regards model generated data, PAMGuide is used as an EDF file writer, since it doesn't perform any processing on the data.

B.1 Installation process

This section intends to guide users through the installation process in both Matlab and Python versions.

B.1.1 PAMGuide installation

PAMGuide was developed in Matlab®, requiring Matlab version 9.3 (R2017b) or higher. To install the PAMGuide package the user should copy the complete folder to the Matlab working directory.

B.1.2 PAM2Py installation

PAM2PY was developed in an open-source platform Python. To install the PAM2PY package the user should copy the complete folder to the PAM2PY working directory. Note that is currently under development the implementation of PAM2PY in the JONAS VRE (Virtual Research Environment) through Jupiter notebooks in JupyterHub.

B.2 Tool packages files

This section describes the Matlab and Python packages available.

B.2.1 PAMGuide package

1. **Graphical User Interfaces (GUI):**

    PAMGuide.fig: main GUI figure that initiate the analysis;

    MetadataExp.fig: experimental metadata GUI figure to input experimental metadata;

    MetadataNum.fig: numerical metadata GUI figure to input numerical metadata;

2. **m-files:**

    PAMGuide.m: main routine to call the PAMGuide GUI interface;

    PG_DFT.m: performs DFT-based analysis for PAMGuide.m;

    PG_Func.m: computes calibrated acoustic spectra from digital audio files;

PG_TOL.m: performs 1/3-octave analysis using the standard filter bank method;
PG_Viewer.m: plots data analysed in PAMGuide.m;
PG_Waveform.m: performs pressure waveform analysis;
prctile.m: calculates the percentiles based on Matlab default routine;
MetadataExp.m: routine to run the MetadataExp GUI interface;
MetadataNum.m: routine to run the MetadataNum GUI interface;
writeh5filenum.m: writes the numerical output in .h5 format;
wirteh5fileexp.m: writes the experimental output in .h5 format;

3. **Example files (in the "Examples" folder):**
   Numeric_File: numerical test file in ".mat" format;
   Numeric_File_2: numerical test file in ".mat" format;
   Sine_10s_48kHz_+-0.5: experimental test file in .WAV format;
   WhiteNoise_10s_48kHz_+-0.5: experimental test file in .WAV format.

B.2.2 PAM2Py package

1. **Graphical User Interfaces (GUI):**
   PAMGuide.PY: main GUI figure that initiate the analysis;

2. **py-files:**
   PAMGuide.py: main routine to call the PAMGuide GUI interface;
   PG_DFT.py: performs DFT-based analysis for PAMGuide.m;
   PG_Func.py: computes calibrated acoustic spectra from digital audio files;
   PG_TOL.py: performs 1/3-octave analysis using the standard filter bank method;
   PG_Viewer.py: plots data analysed in PAMGuide.m;
   PG_Waveform.py: performs pressure waveform analysis;
   write_edf.py: writes the output in .h5 format;
   read_edf.py: read the output in .h5 format;

3. **Example files (in the "Examples" folder):**
   Numeric_data: numerical test file in ".mat" format;
   Sine_10s_48kHz_+-0.5: experimental test file in .WAV format;
   WhiteNoise_10s_48kHz_+-0.5: experimental test file in .WAV format.
   Whistle: experimental test file in .FLAC format.
   DolphinGiggle: experimental test file in .FLAC format.

4. **Others:**
   readme: installing instruction file (IMPORTANT);

B.3 Before running the interactive interfaces

Before running the interfaces, one may want to create a metadata file beforehand (if not, the metadata file will be created during the process through the tool interface). Metadata files have a `.met` extension and may be created and edited using any ASCII file editor.

Two example files are provided in the both packages, "Sine_10s_48kHz_+-0.5_Backup.met" for experimental data and "Numeric_File.met" for numerical data as shown in Figure 13 (a) and (b), respectively. These files are available in the respective example folder. **Note:** No empty spaces should be included after the metadata text.

```
EDF 1.0
Ricardo Duarte
University of Algarve
PT
info@siplab.fct.ualg.pt
20200720
20201212
20201224
Testing
PT-2020-0615-EXP-0001-0010
Experimental
Complete dataset
CM
MarSensing Lda.
SR1-2019
SR-1
Yes
MarSensing Lda.
-185.000000
SR1-2019
SR1
1.000000
20190727
1000.000000
CPC
100
1
40.446000 41.115000
79.982000 81.281000
10.000000
2
1/3-octave-band/base 2
30 minutes
30 minutes
1
7
5 10 25 50 75 90 95
```

(a)

```
EDF1.0
Ricardo Duarte
University of Algarve
PT
info@siplab.fct.ualg.pt
20200720
20201212
20201224
Testing
xxx-xxx-xxx-xxx
Numerical
Complete dataset
AIS Hub
170 120 180
GEBCO
COPERNICUS
COPERNICUS
MacKenzie nine-term equation
Kraken
data_uuidn: xxx-xxx-xxx-xxx
```

(b)

Figure 13: Experimental metadata example file (a) and numerical metadata example file (b).

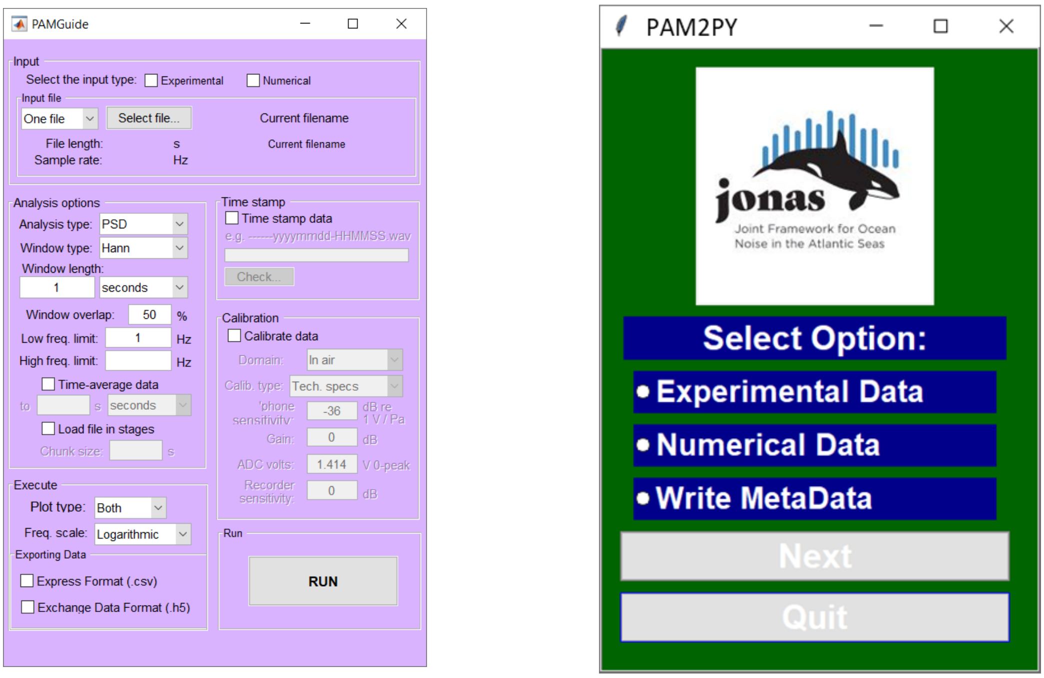
B.4 Running the interactive interfaces

Initiate Matlab or Python according to your convenience and run the respective PAMGuide file, PAMGuide.m for Matlab cases or PAMGuide.py for Python users. The respective graphical interface will automatically pop-up (Figures 14a and 14b).

B.5 Versions panel description

Although the panels are organized differently depending on the version used, the options are similar and allow the same processing. According to this, the following section intends to describe the graphical interface panels.

- **Input**: panel to choose either `experimental` or `numerical` data analysis. In case of experimental two file types are available (.WAV or .FLAC), in case of numerical (.MAT). Then one may select the input type, which can be one file or a complete folder;



(a) (b)

Figure 14: Interactive interfaces of: a) PAMGuide and b) PAM2Py.

- **Analysis options**: this panel integrates the selection of the analysis' type (PSD, TOL, Broadband, Waveform, PowerSpec and TOLf), the type of the Window (Hann, Rectangular, Hamming and Blackman), the window length, the window overlapping and frequency limits. Note that this panel will only be available for the experimental data case;

- **Time stamp**: allows to time stamp the data into the file. Only available in the Matlab version if experimental data is used;

- **Calibration**: allows for the user to introduce the calibration parameters: domain (air or water), hydrophone sensitivity (dB re 1V/Pa), gain (dB), ADC volts (V zero-to-peak) and recorder sensitivity (dB). Only available for experimental data cases;

- **Execute**: allows to select the plot type and the frequency scale. This panel is only available in the case of experimental data;

- **Exporting data**: this panel allows for the user to select the data export file formats. If experimental data is being shared, two exporting formats are available: express format (creates a CSV file) or exchange data format (creates a HDF5 file) which respects the EDF file high level description format shown in Table 9. In the case where model generated data is being exchanged only the EDF file will be available;

- **Run**: this button starts the processing (in case of experimental input) and writes the output data file in the selected format.

B.6 Defining the input numerical file structure required for the model generated data cases

Exchanging model generated data requires the prior creation of a MAT-file or Py-file as an input. These files should contain the following variables:

1. total_number_of_grid_points;
2. longitude;
3. latitude;
4. depth;
5. frequency_band_definition;
6. frequency_count;
7. frequency;
8. time;
9. spl_values;
10. percentile_list.

These variables should follow the variable definition defined in the EDF Table 9.

B.7 Output

The resulting `.csv` and/or the `.h5` files will be created in the same directory of the input file. In the case of a single file being processed, the output file will be created with the same name as the raw (WAV/FLAC) data source file. Considering the processing of a complete folder (batch option), the output file will be created in a folder with the same name but with a "Batch" suffix. Note that if a `.csv` or `.h5` file exists in the same directory with the same name, it will be overwritten.

B.8 Writing EDF through a user provided application

For most experienced users, that use other tools rather than PAMGuide or PAM2Py interactive interfaces, to calculate sound levels this package provide a group of open-source writing and reading functions based on Matlab or Python. The so-called standalone routines are provided in the "Standalone_routines" folder named as:

- write_edf
- read_edf

B.9 Example files

Four testing purpose, test files are supplied in the `Examples` folder in both packages versions. The usage of those files is described in the original PAMGuide description by Merchant et al. [8].

C EDF tool package query

This section intends to guide users in the usage of the EDF tool package by the means of a structured query which should be provided to the support team in case of a debugging is requested.

C.1 Generic information

This section covers the information regarding users operating system and software version (Table 1).

Table 1: Generic information

No.	Action	Answer
1	Indicate your OS	
2	Indicate the software used (Matlab, Phyton or R)	
3	Indicate your software version	
4	Indicate any other relevant particularities	

C.2 Data analysis guided query

This section describes four options provided via PAMGuide/PAM2PY to analyse your data and write it in the EDF file format (Fig. 15). Taking this into account sections C2.1 and C2.2 correspond to the analysis of experimental data using .WAV or .FLAC native file formats. Section C2.1 covers the analysis of a single file, whilst C2.2 deals with the batch situation involving a complete folder containing multiple files.

Then section C2.3 and C2.4 cover the analysis of numerical data considering .MAT file format. Section C2.3 describes how to write an EDF file from one single .MAT file and section C2.4 describes how to create an EDF file from multiple .MAT files.

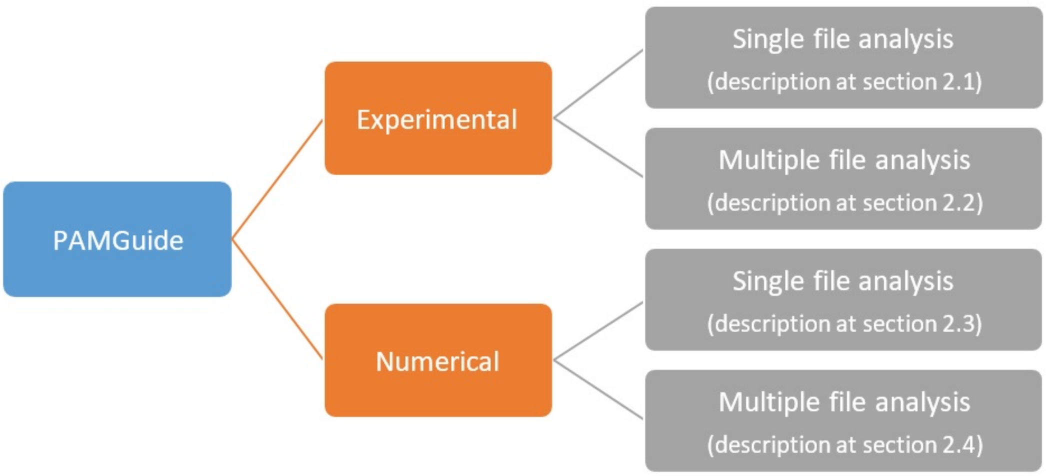


Figure 15: PAMGuide query.

C.2.1 Experimental case – single file analysis

Launch PAMGuide/PAM2PY and select "Experimental". On the file menu consider the "Experimental_example_files" in the "Examples" folder. For testing purpose, this folder contains two WAV files:

- Sine_10s_48kHz_+-0.5.wav
- WhiteNoise_10s_48kHz_+-0.5.wav

For demonstration purposes, only the "Sine_10s_48kHz_+-0.5.wav" file will be considered in the following query.

Table 2: Experimental case - single file analysis

No.	Action	Answer (Y/N)	Comments
2.1.1	Select "Single file" analysis		
2.1.2	In the "Experimental_example_files" folder select "Sine_10s_48kHz_+-0.5"		
2.1.3	Filename on the input panel is: "Sine_10s_48kHz_+-0.5.wav"		
2.1.4	File length: 10s		
2.1.5	Sample rate: 48000Hz		
2.1.6	Select "Exchange Data Format (.h5)"		
2.1.7	Press "RUN"		
2.1.8	Is the metadata pop-up menu displayed?		
2.1.9	Is the metadata pop-up menu completely filled?		
2.1.10	Hold the cursor 3s on each field. Can you see the help string?		
2.1.11	Modify the fields you want		
2.1.12	Let the "comments" field empty		
2.1.13	Press "Export Metadata"		
2.1.14	With your preferred editor open the created ".met" file, and check that the fields correspond to those introduced in the metadata window.		
2.1.15	Check the created EDF file structure using h5disp(' Sine_10s_48kHz_+-0.5.h5')		
2.1.16	Read the EDF file using "read_edf.m" routine provided at the "standalone routine" folder		
2.1.17	Compare the output with "ExpSingleFileAnalysis-h5structure" and "ExpSingleFileAnalysis-plots" files.		

Please Note: Users who may wish to also test the "WhiteNoise_10s_48kHz_+-0.5.wav" file provided in the "Experimental_example_files" folder should take into account that there is no corresponding ".met" file. For this reason, the metadata pop-up menu will be displayed completely empty. See the section C.3 to create the corresponding ".met" file or introduce your metadata directly on the metadata pop-up menu.

C.2.2 Experimental case – multiple file analysis

Launch the PAMGuide EDF tool and tickmark "Experimental Input".

Table 3: Experimental case - multiple file analysis

No.	Action	Answer (Y/N)	Comments
2.2.1	Select "Batch file" analysis		
2.2.2	Select the folder "Experimental_example_files" in the "Examples" folder		
2.2.3	Path name displayed on the input panel is : ". . . /Experimental_example_files"		
2.2.4	Number of files is 2		
2.2.5	Sample rate: 48000Hz		
2.2.6	Select "Exchange Data Format (.h5)"		
2.2.7	Press "RUN"		
2.2.8	Is the metadata pop-up menu completely empty?		
2.2.9	Hold the cursor 3s on each field. Can you see the help string?		
2.2.10	Modify the fields you want		
2.2.11	Press "Export Metadata"		
2.2.12	With your preferred ASCII editor open the created ".met" file, and check that the fields correspond to those introduced in the metadata window.		
2.2.13	Check the created EDF file structure using h5disp(' PAMGuide_Batch_Experimental _example_filesFolder.h5')		
2.2.14	Read the EDF file using "read_edf.m" routine provided at the "standalone routine" folder		
2.2.15	Compare the output with "ExpMultipleFileAnalysis-h5structure" and "ExpMultipleFileAnalysis-plots" files.		

Please Note: No ".met" file is provided for batch file analysis case. For this reason, the metadata pop-up menu will be displayed completely empty. See the section C.3 to create the corresponding ".met" file or introduce your metadata directly on the metadata pop-up menu.

C.2.3 Numerical case – single file analysis

On the PAMGuide interface tick "numerical input" box. On the file menu consider the "Numerical_example_files" in the "Examples" folder. This folder contains two .MAT files for testing:

- Numeric_File.mat
- Numeric_File2.mat

For demonstrative purposes, only the "Numeric_File.mat" file will be considered in the following query.

Table 4: Numerical case - single file analysis

No.	Action	Answer (Y/N)	Comments
2.3.1	Select "Numeric_File.mat"		
2.3.2	Filename on the input panel is: "Numeric_File.mat"		
2.3.3	Select "Exchange Data Format (.h5)"		
2.3.4	Press "RUN"		
2.3.5	Is the metadata pop-up menu displayed?		
2.3.6	Is the metadata pop-up menu completely filled?		
2.3.7	Hold the cursor 3s on each field. Can you see the help string?		
2.3.8	Modify the fields you want		
2.3.9	Press "Export Metadata"		
2.3.10	With your preferred ASCII editor open the created ".met" file, and check that the fields correspond to those introduced in the metadata window.		
2.3.11	Check the created EDF file structure using h5disp('Numeric_File.h5')		
2.3.12	Read the EDF file using "read_edf.m" routine provided at the "standalone routine" folder		
2.3.13	Compare the output with "NumSingleFileAnalysis-h5structure" file.		

Please Note: In case you want to test also the "Numeric_File2.mat" file provided at the "Numerical_example_files" folder take into account that there is no corresponding ".met" file. For this reason, the metadata pop-up menu will be displayed completely empty. See the section C.3 to create the corresponding ".met" file or introduce your metadata directly on the metadata pop-up menu.

C.2.4 Numerical case – batch file analysis

Launch PAMGuide EDF tool and tickmark “Numerical Input”

Table 5: Numerical case - single file analysis

No.	Action	Answer (Y/N)	Comments
2.4.1	Select “Batch” file analysis		
2.4.2	Path name displayed on the input panel is :“. . . /Numerical_example_files”		
2.4.3	Select “Exchange Data Format (.h5)”		
2.4.4	Press “RUN”		
2.4.5	Is the metadata pop-up menu displayed?		
2.4.6	Is the metadata pop-up menu completely empty?		
2.4.7	Hold the cursor 3s on each field. Can you see the help string?		
2.4.8	Modify the fields you want		
2.4.9	Press “Export Metadata”		
2.4.10	With your preferred ASCII editor open the created ”.met” file, and check that the fields correspond to those introduced in the metadata window.		
2.4.11	Check the created EDF file structure using h5disp(‘Numerical_example_files_Batch_Num.h5’)		
2.4.12	Read the EDF file using “read_edf.m” routine provided at the “standalone routine” folder		
2.4.13	Compare the output with “NumMultipleFileAnalysis-h5structure” file.		

Please Note: No “.met” file provided for batch file analysis. For this reason, the metadata pop-up menu will be displayed completely empty. See the section C.3 to create the corresponding “.met” file or introduce your metadata directly on the metadata pop-up menu.

C.3 Creating and loading your own metadata file

Using your own files requires corresponding metadata. This section provides guidance on the metadata file creation process.

Table 6: Writing and reading metadata

No.	Action	Answer (Y/N)	Comments
3.1	Taking into account the format described at the "Short User Manual for the Exchange Data Format package" open your preferred ASCII editor and create all metadata fields OR use a reference file ("Sine_10s_48kHz_+-0.5.met" or "Numeric_File.met") file according to your data input type (experimental or model generated data respectively) and modify the fields at will		
3.2	Save it with ".met" extension		
3.3	Repeat the steps of the section C2.1 or C2.2 in case of experimental input or sections C2.3 or C2.4 in case of numerical input		
3.4	In case your ".met" file was saved with a different name of your input file press "Load Metadata file" and select the metadata file you created		
3.5	Are the metadata fields correctly completed?		
3.6	If not, revise your ".met" file and repeat the procedure.		
3.7	If the error persists, indicate the erroneous field(s) and attach the your files to the support team at info@siplab.fct.ualg.pt.		

C.4 Standalone routines

Standalone routines are available on the "Standalone_routines" folder.

Important notes: The fields corresponding to the numerical models should be empty in the case of experimental data and vice-versa.

Table 7: EDF writing

No.	Action	Answer (Y/N)	Comments
4.1.1	Select the "Standalone_routines" directory		
4.1.2	Type "help write_edf" in order to see the description of this function		
4.1.3	Create your ".h5"		
4.1.4	Check if the ".h5" file was successfully created in the working directory		
4.1.5	Type "h5disp('filename.h5')" to evaluate the file structure		
4.1.6	Correct datasets?		
4.1.7	Correct attributes?		
4.1.8	Use "h5read" to check if the data in the ".h5" file is according to what was introduced		
4.1.9	Describe any other possible problems that was encountered		

Table 8: EDF reading

No.	Action	Answer (Y/N)	Comments
4.2.1	Select the "Standalone_routines" directory		
4.2.2	Ensure that the ".h5" file is in the same directory.		
4.2.3	Type "read_edf('filename.h5')"		
4.2.4	Is it possible to open the ".h5" file?		
4.2.5	Are all the fields displayed in the screen?		
4.2.6	Correct datasets?		
4.2.7	Access to the output structure and confirm the content of the fields		

C.5 Other important feedback

Please list here any other remarks and suggestions you have regarding the tool. Try and be as specific as possible, giving consideration to features or aspects of the work flow that you feel could be revised or improved.

Having repeated the prescribed work flow procedures with your own files, you are now invited to list below any issues encountered, specifying, in the case of experimental input, the type of the file processed (WAV/FLAC).

Note: In case of using numerical input, it is important to ensure that you first prepare your numeric file according to the “Short User Manual for the Exchange Data Format package”.

D JONAS Exchange Data Format proposal

Table 9: JONAS Project - Action 4.3: Exchange data format proposal using HDF5 file format.

Object name	Object field	Required	Object Type	Dimension	Field definition	Example
/	group				General description of the field	
format_version	attribute	Yes	string(10)		Format specification version	e.g."EDF1.0"
author	attribute	Yes	string(30)		Creator of the HDF5 file	e.g."Ricardo DUARTE"
institution	attribute	Yes	string(50)		Data provider Institution	e.g."University of Algarve, SIPLAB"
country_code	attribute	Yes	string(5)		Country and region codes according to ISO 3166	e.g."PT for Portugal"
contact	attribute	Yes	string(50)		Contact for all external queries in the future	e.g."e-mail:rjduarte@ualg.pt"
start_date	attribute	Yes	integer(20)		Data when the analysis started according to ISO 8601	e.g."20190131 for the 31st of January 2019"
end_date	attribute	Yes	integer(20)		Data when the analysis ended according to ISO 8601	e.g."20190231 for the 31st of February 2019"
date_of_creation	attribute	Yes	integer(20)		File creation date according to ISO 8601	e.g."20190310 for the 10th of March 2019"
purpose	attribute	Optional	string(500)		Objective of the experimental/numerical analysis	e.g."Evaluation of the underwater noise at Azores archipelago included in JONAS project"
data_uuid	attribute	Yes	string(30)		Data unique identification number, linking exchange data with raw data using: "country_code-year-month_day-datatype-file_number/total_files	e.g."PT-2020-0615-EXP-0001-0010"
data_type	attribute	Yes	string(20)		Indication whether is experimental, numerical or combined data	e.g."Combined"
comments	attribute	Optional	string(300)		Generic observations	e.g."Complete dataset"
/analysis_metadata	group				General description of the file metadata	
/experimental	group	Yes			General description of the experimental metadata	
setup	attribute	Optional	string(20)		Description of the deployment: Autonomous - AUT; Cable Mounted - CM; Combined - COMB; Bottom Frame - BF; From Vessel - FV; Glider - GLD Mooring with floating buoy - MFB; Other - OTH	e.g."CM" and "BF"
/recorder	group				General description of the recorder	
recorder_manufacturer	attribute	Optional	string(100)		Recorder manufacturer	e.g."MarSensing Lda."
recorder_serial_number	attribute	Optional	string(100)		Recorder serial number	e.g."SN45736"
recorder_model	attribute	Optional	string(100)		Recorder model	e.g."SR-1"
builtin_hydrophone	attribute	Optional	string(5)		Recorder and the hydrophone are one	e.g."Yes"
/hydrophone	group				General description of the hydrophones	
hydrophone_manufacturer	dataset	Optional	string(100)	hydrophone_count	Hydrophone manufacturer	e.g."MarSensing Lda."
hydrophone_sensitivity	dataset	Yes	float(10)	hydrophone_count	Hydrophone sensitivity provided by the manufacturer in $dBre1V/\mu Pa$	e.g."-185"
hydrophone_serial_number	dataset	Optional	string(100)	hydrophone_count	Hydrophone serial number	e.g."SN-SR1-2019-2"
hydrophone_model	dataset	Optional	string(100)	hydrophone_count	Hydrophone model	e.g."SR1"
/calibration	group				General description of the equipment calibration for each hydrophone	
calibration_frequency_count	dataset	Yes	integer(2)	hydrophone_count	Number of frequencies used to calibrate hydrophones	e.g."1"
calibration_datetime	dataset	Yes	integer(20)	hydrophone_count	Date and time when the system was calibrated according to ISO 8601	e.g."20180101 for the 1st of January of 2018"
calibration_factor	dataset	Yes	float(5)	hydrophone_count x calibration_frequency_count	Factor to convert raw data from volts to dB re $1\mu Pa$	e.g."1000"
calibration_procedure	dataset	Yes	string(300)	hydrophone_count	Procedure used to calibrate hydrophones according to ICES codes. See the entire ICES code list at: "vocab.ices.dk/?ref=1591"	e.g."CPC"
reference_frequencies	dataset	Yes	float(5)	hydrophone_count x calibration_frequency_count	Calibration frequencies in Hz	e.g."100"
/numerical_model	group	Yes			General description of the numerical model metadata	
ais_database	attribute	Yes	string(100)		Description of the AIS database used in the numerical models.	e.g."AISHub - www.aishub.net"
source_levels	dataset	Yes	integer(10)		Description of the emitted source levels in dB by vessel category according to their AIS ship type code First line indicating the ship type and second line the corresponding source level	e.g."[50 60 80; 170 120 180] "
bathymetry_database	attribute	Yes	string(100)		Description of the bathymetry database	e.g."GEBCO - www.gebco.net"
temperature_database	attribute	Yes	string(100)		Description of the water column temperature database	e.g."COPERNICUS - www.copernicus.eu"
salinity_database	attribute	Yes	string(100)		Description of the water column salinity database	e.g."COPERNICUS - www.copernicus.eu"
sound_speed_profile_model	attribute	Yes	string(100)		Description of model used to calculate the sound speed profile	e.g."K.Mackenzie-nine-term equation"
propagation_model	attribute	Yes	string(10)		Description of the propagation model	e.g."Kraken"
numeric_data_calibration	attribute	Yes	string(50)		Description of the experimental data file used to calibrate numerical models.	e.g."data_uidn:xxxx-xxxx-xxxx-xxxx"
/ambient_noise_dataset	group	Yes			General description of ambient noise	
/position	group	Yes			General description of the location	
hydrophone_count	dataset	Yes	integer(2)		Number of hydrophones	e.g."1"
total_number_of_grid_points	dataset	Yes	integer(10)		Total number of points in the numerical grid	e.g."1000"
longitude	dataset	Yes	float(10)	hydrophone/grid_count x time_count	Longitude coordinates vector in decimal degrees	e.g."[40.446,...,41.115]"
latitude	dataset	Yes	float(10)	hydrophone/grid_count x time_count	Latitude coordinates vector in decimal degrees	e.g."[79.982,...,81.281]"
depth	dataset	Yes	float(10)	hydrophone/grid_count x time_count	Depth vector according to the mean reference sea level in meters	e.g. "10"
/frequency	group	Yes			General description of ambient noise frequencies	
frequency_count	dataset	Yes			Number of frequencies	e.g."2"
frequency_band_definition	dataset	Yes	integer(10)	frequency_count	Describes the frequency band used and base	e.g."1/3-octave-band in base 2"
frequency	dataset	Yes	float(5)	frequency_count	Frequencies in Hz at which the SPL is calculated	e.g."63" and "126"
/time	group	Yes			General description of ambient noise time	
time_duty_on	attribute	Yes	integer(10)		Duration that the device is recording	e.g."30 minutes"
time_duty_off	attribute	Yes	integer(10)		Duration that the device is not recording. If the device is always recording type "0"	e.g."30 minutes"
time_count	dataset	Yes	float(10)		Number of time entrances since start_datetime	e.g."1250"
time	dataset	Yes	float(10)	time_count	Time vector in POSIX time seconds	e.g."[1562151105,...,1562153100]"
/sound_pressure_levels	group	Yes			General description of sound pressure levels (spl)	
averaging_time	attribute	Yes	integer(5)		Averaging time in seconds	e.g."1"
spl_values	dataset	Yes	float(10)	hydrophone/grid_count x frequency_count x time_count	SPL over time for all covered frequency bands in dB re $1\mu Pa^2/Hz$ Using: $SPL(f) = 10log_{10}\left(\frac{P_{ss}}{B\Delta f^2 p_{ref}^2}\right) - S(f)$	e.g."[25,...,30]"
/sound_pressure_levels_stats	group	Yes			General description of sound pressure levels statistics	
percentile_count	dataset	Yes	integer(2)		Number of percentiles	e.g."7"
percentile_list	dataset	Yes	integer(50)	percentile_count	List of percentiles	e.g. "5,10,25,50,75,90,95"
percentile_values	dataset	Yes	float(10)	hydrophone/grid_count x frequency_count x percentile_count	SPL considered percentiles in [dB re $1\mu Pa^2$]	e.g."[74,...,81]"

Revision History

Date	Version	Description	Author
17/03/2021	0.1	First version - draft	Ricardo Duarte
12/07/2021	0.2	First version comments - draft	Sérgio M. Jesus
22/07/2021	1.0	First version	Ricardo Duarte
28/09/2021	1.0	First Revision	José Diaz and Eric Delory
11/10/2021	2.0	Second version	Ricardo Duarte
19/10/2021	2.0	Second version comments	Sérgio M. Jesus
22/10/2021	3.0	Third version	Ricardo Duarte