**Requirements for SEATRACK database**

**What is needed?**

It has become clear that the project needs a more comprehensive database. Our current data-storage systems and methods are not viable in the long-run, especially not if we intend to incorporate into SEATRACK other types of data and/or similar data from other tracking projects. Our current processes are slow and error prone and in some cases overly complex. We need to have more automated processes importing metadata, adding additional data afterwards (such as results of sexing) or changed when an error is discovered without having to go through several excel sheets to make sure an error has been eradicated.

The general structure of the data we have available is, excluding redundancy, pretty simple and allows us to design a database around it. Most or all current tables could be broken down into their base components and should be linkable (or have already been linked) using existing keys such as individual ids, logger ids and relevant timings. Afterwards better non-changing unifying keys can be designed and the processes automated.

A new database would ideally include a more automatic processes for storing and keeping track of rawfiles, automatically naming and sorting such files in a more simple and streamlined fashion than in the current dropbox folder setup. Thereby reducing the risk of human error and making it quicker and easier to access and use such data.

The post doc and next large data input session start in the fall, therefore we would ideally like to have a testable version in place by then. This work needs to coincide with the setup of a common server.

In principle we would like a relational database that:

* Allows us to combine all data-streams.
* Streamlines data-storage, i.e. reduces redundancy in data. Single source when possible.
* Allows multiple users to work with the same data, at the same time from different locations.
* Is easy to use, e.g. has user-friendly interfaces for diverse purposes, e.g. importing metadata, processing, startup, “logistics” and for making queries.
* Links data at all stages of processing to the raw files.
* Contains database log – activity log for database.
* Is machine searchable.
* Has clear export options and protocols.
* Contains clear description of all data and processes; values, resolution, limitations, methodology etc…

**Current status of data setup and data availability.**

Currently SEATRACK maintains a positional database, mainly in a single table in which the primary key is a timestamped position of an individual. Other variables included in the database are imported from so called “**processing files**” which are excel files created during the processing of light data. A processing file is created for any given species at any given location, each year one or more loggers are retrieved. The primary key is “logger\_yeartracked” but this type of file holds around 84 variables of which 34 are directly relevant to all SEATRACK data of which around 7 can be called unique to this table, mostly start and end dates of certain periods within a “logger year” but also sunangle. The other variables are either copies of variables, mainly from the so called “**metadatafiles**” or can be derived there from. These are manually copied between these two excel-files when the logger is being processed. At that point in time it is not always clear if the data being copied is correct and it might need to be changed later on.

The “metadatafiles” are excel files structured as forms filled out by project participants after the field-season has ended. The primary key is a trapping event or in few cases an observation. Each trapping event can include a retrieval of a logger, a deployment of a logger, both or neither in the case of an observation. These metadata files structured into eight sub-tables:

* Ringing info, contains information about the rings /ids of an individual. Note that metal-ring ID can be changed for an individual and cannot by itself be used as an individual ID.
* Logger info, contains info about the logger/s being retrieved, deployed or both or observed.
* Individual information, contains info pertaining to an individual, species, sex morph etc. that do not/should not change through time (age\*).
* Morphometrics, contains certain measurements of an individual at the time of encounter.
* Breeding status, contains information pertaining to observed breeding output at the time of the encounter and in some cases breeding success later on.
* Breeding location, contains information about the locality an individual is trapped. Note that currently the most commonly used version of this information is colony from the colony\_table which is not necessarily the value shown in the metadata field but contains it within in colony table.
* Samples, contains information about samples taken from the individual and availability thereof. For the most parts very standardized and simple.
* Other, contains variables that could not be fitted into the other categories the most important of which being “data responsible” and a comment field.

It varies between participants how many metadatasheets they turn in, some few turn in metadata sheets for each species at their colony while some others combine several species at more than one colony in the same metadata sheet. Most do however turn in a single metadata sheet for all species in a given colony for a given year.

Although there are quite strict and simple instructions on how to fill out these sheets the quality of this work varies greatly amongst participants so a lot of work and time goes into error checking the data, which currently is done manually to a large degree. When errors are discovered, participants are asked to help fix errors. This can take a long time and sometimes results in a new metadata file being turned in with several changes. In the meantime data can have been processed and flawed data copied into the processing tables can therefore end up in the current database.

Due to the nature of collecting logger-data, information pertaining to a single logger\_session (Figure 1) will in almost all cases come from two separate metadatasheets, metadata from deployment occasion (t0) and metadata at retrieval occasion (t1+x) usually one or more years later. The subsequent structure of the data does therefore not allow direct linking of events within the same logger\_session. For this purpose and others we have constructed the so called “**loggeroverview**” excel file.



Metadata t0

Metadata t1+x

Figure 1 A flowchart displaying any possible event and/or route a logger might take throughout a logging session, from startup to being decommissioned or restarted for another logging session. After startup a logger gets assigned or allocated to a certain species at a certain colony, after that it might either be deployed or shut down (sent to sleepmode). If shut down it will either be stored at location or sent back to SEATRACK. In either case it will either be restarted and reused or be decommissioned depending on age and or loggermodel. If the logger was deployed we will receive info on the deployment through metadata and the logger will hopefully record information while being at large. Until it is retrieved or the individual observed having lost the logger no information will be added to the logger session and it will remain open/active. If retrieved information pertaining to that event are included in the metadatasheet for that year, most commonly 1-2 years later. The raw data are then downloaded if possible, if not the logger is sent to the logger-producer for data-reconstruction. What kind of data are downloaded, and in what form they are depend on logger make and model but are still very similar in structure. After a successful download some loggers are restarted and reused while others, depending on model and age, are decommissioned. All loggers that are sent to producers are considered decommissioned.

The logger-overview file contains 51 columns arranged into six sub tables where the primary key is the logger session, or more accurately, loggerID\_yearstarted. In almost, if not, all cases year started equals year deployed. The subtables are organized as follows:

* Startup, contains information about the startup of the logger, e.g., time, date, startmode and who started it and where as well as some basic info on the logger itself, such as make, model and production year.
* Assignment/allocation, contains information about where the logger was/is being sent and for what purpose, i.e intended species, location and receiver. On what species the logger finally ends up is a different matter though.
* Deployment, contains basic data extracted from metadatasheets (t0).
* Retrieval, contains basic data extracted from metadatasheets (t1+x).
* Download/shut down, contains basic information about the downloading/stopping process, i.e. when was it sent to sleepmode and where, was the download successful or needs to be/was reconstructed and is the file available?
* General, contains several variables that actually could be included in some of the subtables or contains information copied or inferred from metadatafiles or “**Loggerstores**”. Most important of these are:
  + Logger fate was a logger deployed or returned/stored,
  + Logger endstatus was it retrieved / and redeployed, lost or simply still at large.
* Other variables within this subtable contain info on whether a logger was sent to producer, if it can be redeployed after session ended and to which project the logger actually belongs which in almost all cases is SEATRACK.

As is the overview file is used for multiple purposes such as extracting data on deployment and retrieval successes between colonies, species and years, to keep track of loggers available throughout the logger session, e.g. how many are at large, how many are available for deployment next year, how many of this model/production year have failed and how many are being sent to producers. Its main purpose though is to check for errors and inconsistencies, e.g. was the logger actually retrieved from the same bird it was deployed on, does this loggerID exist, was it allocated to this participant etc. All of these inference-type variables will become redundant when a relational database in place if setup correctly and completely.

**Loggerstores** is an excel file that contains information about loggers having been returned to SEATRACK, where they came from and why (retrieved, not used or failed) and what is to be done with them, restart and redeploy decommission, or send to producer. All of this info is contained as well in loggeroverview and can there for be ignored for now. But the database should contain such a table.

Other tables exist and are used, mostly intended to keep track of certain descriptive terms such as the colony table but more are missing describing processes, methodology, datatypes, etc.

**RAW data**

Currently the raw data is stored and sorted in to folders based on locations, processing status, species and year. The current naming convention of files is LoggerID(reconstructed or not?)\_YearRetrieved\_LoggerModel but there are some inconsistencies in how these are named/written as it is all done manually. What type of data is downloaded varies between loggermodels in some cases within logger loggermodels based on how these were programmed to begin with (Loggingmode). Broadly loggers can be divided by producer[[1]](#footnote-1) and size, biotrack (BT) large (mk3006 and mk3005), Biotrack small (mk4083 and mk4093), migratetech (MT) large (C330 and C250) and migratetech small (C65, W65 and F100). Large models usually record temperature at different intervals (so does F100 and depending on the logging-mode also C65) but otherwise all of these models record activity (wet/dry state counts) and most importantly light levels. There are several files created during the processing/extracting such data (Figure 2) but in **broad** terms these can be classified as .LIG (MT) and .LUX (BT) contain timestamped light level values, .DEG (MT) .ACT(BT) contain timestamped activity data (wet counts) and .SST (MT) and .TEM (BT) contain timestamped temperature readings. During processing of the .LIG and .LUX files are converted into transition files, .TRN which contains timestamped twilight events, i.e. either a sunrise or sunset from which noon and midnight are inferred. These events are furthermore assigned a confidence value based on the expected/observed quality of the reading. The next step is calculating raw positions based on the timings of these events and estimated sunangle wich is then stored in a trajectory file, TRJ. These raw positions are then averaged (smoothed) twice and both smoothed and double smoothed positions along with the timestamped raw positions and confidence values are what make up the most important part of the current database.

Migratetech

Biotrack



Figure 2 A flowchart showing the types raw files downloaded from a migratetech loggers on the left and Biotrack loggers on the right and those created automatically during processing. Currently the database only keep track of the positional data calculated from the .lux and .lig files.

Although the data recorded is structured pretty similarly between the logger producers there are differences which need to be included in the description tables. Not only that, but Migratetech loggers actually contain a lot of information about each individual logger and logger session in their memory. Although some of this is actually contained in other tables (e.g. tables containing startup and logger info) this is the most secure data source and it should be considered if, when possible (only for migratetech loggers) this data should be read into the relevant tables in the database.

**Suggested setup and next steps**

Given the current setup we have in place and the general structure of the data and processes (Figure 3) we have envisioned a relational database roughly based on several tables, similar in structure to our current ones, and a raw data file archive. The setup would be based on several unifying keys most important ones being logger\_unique\_ID, Individual\_unique\_ID and Session\_ID, but also event\_IDs, file\_IDs etc.

The main tables being:

* Individual logger Info, containing basic info relating the logger itself, such as make, model and production year. Primary key = **unique\_loggerID.** Current source = logger overview.
* Startup, contains information about the startup of the logger, e.g., time, date, logging mode and who started it. Primary key/join = **unique\_loggerID+timestamp or SessionID.** Current source = logger overview. Note that each logger can have several startups/sessions but only a single valid one per year.
* Assignment/allocation, containing information about where the logger was/is being sent and for what purpose, i.e intended species, location and receiver. Primary key/join = **unique\_loggerID+timestamp or SessionID.** Current source = logger overview. The importance of this table is for querying where logger can be found and for error reading data.
* Bird-table contains info about individual birds that does not change after the initial encounter. Primary key **unique\_individualID.** Note that each individual can have several encounters and logging sessions during its life, albeit only a single deployment and retrieval per logging session.
  + Sex contains information of the sex of an individual as well as metod used for sexing and time of said sampling/measuring used for determination. Although sex is a constant for each individual samples/measurements can be taken more than once and in few cases yield different results so in praxis it might be better to include sex as a separate table rather than a variable in birdtable?. Not only that but sex analysis often take time or are carried out years after sampling events have taken place so such data need to be easily imported and linked up to related data in the database.  
    Primary key/join = **unique\_individualID+timestamp**
* Trapping events / observations contains date, time and location of an event and the unique individual id of the individual involved. It needs to be decided whether it is best to keep a single table like this for all events or split it up in three based on encounter type. In that regard it is worth noting that in many cases retrieval and a subsequent deployment coincide, i.e. the end date/event of one logging session is the same as the start of another. It should also be decided whether or not the following data should be kept in separate tables or included as they are today in a single large table. Primary key/join = **Unique\_individualID + loggerID + timestamp or SessionID or some combination thereof.** 
  + Morphometrics, contains certain measurements of an individual at the time of encounter. Could possibly be included in the trapping events / observation table or stand alone, linked with individual unique ID + timestamp.
  + Samples, contains information about samples taken from the individual and availability thereof. For the most parts very standardized and simple. Could possibly be included in the trapping events / observation table or stand alone, linked with individual unique ID + timestamp.
  + Breeding status, contains information pertaining to observed breeding output at the time of the encounter and in some cases breeding success later on. Could possibly be included in the trapping events / observation table or stand alone, linked with individual unique ID + timestamp.
  + Descriptive tables such as the colony table, species lists etc. are not included in this since it needs to be discussed whether they should be included in their entirety within any of these tables or separate, linked with an identity key. For compatibility with existing databases, it is advisable to implement existing widespread universal codes such as EURING when possible.
* Download/shutdown (end of logging) contains basic information about the downloading/stopping process, i.e. when was it sent to sleep mode and where, was the download successful or needs to be/was reconstructed and is the file available?
* Loggerstores contains information about loggers having been returned to SEATRACK, where they came from and why (retrieved, not used or failed) and what is to be done with them, restart and redeploy decommission, or sent to producer. Should largely be based on functions drawing on data from other tables such as allocation, end of logging, deployments and retrievals.
* Processing table consists of those variables unique and necessary to the processing. Sunangle, start and end of periods of midnightsun/polarnight effects and equinoxes. Key attribute equals or is equivalent to LoggerID+year\_logged.
* RAW-data archive contains a list of file-names and storage\_locations and other info about all rawfiles in the archive. Automatic naming system and automatic entry?
* Positional\_data\_table, contains both smoothed and double smoothed positions along with the timestamped raw positions and confidence values, the most important part of the current database. Primary key is a timestamped position of an individual.
* Post-processing product archive, contains a list of names, storage-locations/links and other info about all maps, kernels, animations and (publications?).

This list is not exhaustive or intended to be a final version but only meant to serve as an outline of what is requested. As stated earlier, the idea is to use the general structure of the data we have available and design a database around it. Most or all current tables could be broken down into their base components and should be linkable using existing keys such as individual ids, logger ids and relevant timings. It is however necessary to come up with better non-changing unifying keys in the long run. Much of the data in these tables are however redundant and a lot of the tasks in relation to data storing and processing unnecessary complicated and error prone and could be replaced by automatic functions and queries.

**Interfaces** need to be user-friendly and **queries, imports and exports** easily implemented. Interfaces for entering data on individual logger basis, such as during startup, processing, download and shutdown as well as viewing and correcting data. Ideally metadata could be imported directly on error checked to a large degree automatically.



Figure 3 A flowchart showing the general structure of data and processes. The main component sources are indicated by colors; bird info (green), logger info (blue), processing (orange) and post processing products (purple).

Appendix

A rough description of values in current main tables and their sub-tables.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table** | **Subtable** | **Column\_name** | **Type** | **Explanation** |
| Loggeroverview | General | Rown | number | Rownumber |
| Loggeroverview | General | Row\_OK | Binomial | 1 = OK, 0 = missing/suspicous data |
| Loggeroverview | General | Deployment\_completed | Binomial | 1 = Logger in hand, 0 = Logger in deployment (Note that loggers stored are said to have completed deployment weather they were deployed or not) |
| Loggeroverview | Startup | LID | string | Logger ID |
| Loggeroverview | Startup | start\_gmt\_date | Date | Date when logger was started acc. GMT time |
| Loggeroverview | Startup | start\_gmt\_time | Time | GMT Time when logger was started |
| Loggeroverview | Startup | logging\_mode | list | Needs to be adjusted with regards to different loggertypes, producers, projects and what modes have been used so far |
| Loggeroverview | Startup | producer | list | Name of producer [migratetech;biotrack;staroddi;...] |
| Loggeroverview | Startup | type | list | Name of loggermodel [mk3006;mk3005;mk4083;mk4093;c250;c65;F100;w65...] |
| Loggeroverview | Startup | prod\_year | Factor | Year logger was produced if known |
| Loggeroverview | Startup | Deployment\_session | Factor | Year logger was deployed or intended to be deployed |
| Loggeroverview | Startup | Ret\_year | Factor | Year logger was retrieved |
| Loggeroverview | Startup | started\_by | list | Who started logger, if known [VB;HHH;BM;EL;...] |
| Loggeroverview | Startup | started\_where | list | Location were logger was started, if known [Tromsø;Trondheim;Jan Mayen;Bjørnøya;...] |
| Loggeroverview | Startup | Days delayed (0 = start now) | number | Number of days iniation of start of logging has been delayed |
| Loggeroverview | Startup | programmed\_gmt\_date | Date | Date when logger was intended to start after delayed initation of logging acc. GMT time |
| Loggeroverview | Startup | programmed\_gmt\_time | Time | GMT Time when logger was intended to start after delayed initation of logging |
| Loggeroverview | Assignment | intended\_species | list | what species was the logger intended for (Only important for SEATRACK loggers) |
| Loggeroverview | Assignment | intended\_locality | list | where were the loggers intended to be deployed (Only important for SEATRACK loggers) |
| Loggeroverview | Assignment | intended\_deployer | list | Who was intended as the deployer (Only important for SEATRACK loggers) |
| Loggeroverview | Assignment | comment | string | Comment1 refers to start or original intentions (office plan) |
| Loggeroverview | Deployment | LoggerID2 | string | Logger ID, does not need to be repeated but good to have when merging data from different files to starting info |
| Loggeroverview | Deployment | Dep.dataOK | Binomial | 1 = OK - all info about deployment adds up, 0 = missing/suspicous data |
| Loggeroverview | Deployment | IndID | string | If possible EURINGCODE\_METALNR. |
| Loggeroverview | General | Loggerfate | list | What happened to the logger [Deployed;Stored;Lost;...] |
| Loggeroverview | General | LoggerEndstatus | list | How did the logger endup [Retrieved;Still at large; Sleeping (stored); Logging (stored);lost;...] |
| Loggeroverview | Deployment | Dep. Species | list | What species was the logger deployed on |
| Loggeroverview | Deployment | Dep.Date | Date | When was the logger deployed |
| Loggeroverview | Deployment | Dep.colony | list | Where was the logger deployed according to SEATRACK Colony list |
| Loggeroverview | Deployment | Dep.loc | string | Where was the logger deployed as indicated in metadatasheet |
| Loggeroverview | Retrieval | DELETE | string | Logger ID |
| Loggeroverview | Retrieval | Retrival-type | list | How was the logger retrieved, [Retrieved breeding; Retrieved out side of breeding season; ...] |
| Loggeroverview | Retrieval | Retrievaldate | Date | When was the logger retrieved |
| Loggeroverview | Download | Downloadtyp | list | How was the logger downloaded [Successfull; Reconstructed; Failure;...] |
| Loggeroverview | Download | File\_Available | Binomial | 1 = File available in data storage or has been imported to database, 0 = File ruined or not available for any reason |
| Loggeroverview | Download | Stop/download/failed date | Date | When was the logger downoaded/sent to sleep |
| Loggeroverview | Download | Downloadtime | Time | What time was it downloaded |
| Loggeroverview | Download | Comment2 | string | Comment2 refers to deployment, retrieval and/or download |
| Loggeroverview | General | Ready for redeployment | Binomial | 1 = Can be redeployed, 0 = It cant (This has not yet been fully implemented in the file.) (2 = has allready been redeployed) |
| Loggeroverview | General | Logger returned to BT | Binomial | 1 = Logger has been returned to biotrack to be replaced, 0 = it hasn't |
| Loggeroverview | General | Available for depl. Or repl.in 2014 | Binomial | 1 = Logger was available for deployment in 2014 |
| Loggeroverview | General | Available for depl. Or repl.in 2015 | Binomial | 1 = Logger was available for deployment in 2015 |
| Loggeroverview | General | Available for depl. Or repl.in 2016 | Binomial | 1 = Logger was available for deployment in 2016 |
| Loggeroverview | General | Available for depl. Or repl.in 2017 | Binomial | 1 = Logger was available for deployment in 2017 |
| Loggeroverview | General | Project | list | For what project were the loggers deployed [SEATRACK;SEATRACKpilot;TheGuillemotProject;TOTAL;...] |
| Loggeroverview | Retrieval | checkmet | NA | Quality check, deployed = retrieved |
| Loggeroverview | Retrieval | Checkloc | NA | Quality check, deployed = retrieved |
| Loggeroverview | Retrieval | checkspec | NA | Quality check, deployed = retrieved |
| Loggeroverview | General | Where? | NA | Quality check, deployed = retrieved |
| Loggeroverview | General | LOGGcheck | NA | Quality check, deployed = retrieved |
| Loggeroverview | General | metcheck | NA | Quality check, deployed = retrieved |
| Metadata16 | Ringing info | date | Date | [dd.mm.yyyy] |
| Metadata16 | Ringing info | ring\_number | string | metal ring number e.g. '6294566', 'CA22632', '39445432' |
| Metadata16 | Ringing info | euring\_code | list | Code for ringing central providing rings: 'NOS' -Norway (Stavanger),'NOO' -Norway (Oslo), 'ISR' Iceland (Reykjavik), 'RUM' - Russia (Moscow), 'DKC'-Denmark (Copenhagen) |
| Metadata16 | Ringing info | color\_ring | string | enter colour ring code e.g. 'J3215', 'ABB' etc |
| Metadata16 | Logger info | logger\_status | list | logger status if individual is caught, observed or found dead |
| Metadata16 | Logger info | logger\_model\_retrieved | list | pick model from drop-'down list |
| Metadata16 | Logger info | logger\_id\_retrieved | string | enter the logger id of the logger retrieved e.g. '22314', 'v2009001', 'n054' |
| Metadata16 | Logger info | logger\_model\_deployed | list | pick model from drop-'down list |
| Metadata16 | Logger info | logger\_id\_deployed | string | enter the logger id of the logger deployed e.g. '22314', 'v2009001', 'n054' |
| Metadata16 | Individual information | species | list | Pick species from drop down list. |
| Metadata16 | Individual information | morph | string | if relevant, mention morph, e.g. bridled or non-bridled |
| Metadata16 | Individual information | subspecies | list | if relevant, mention subspecies, e.g. Fuscus intermedius, fuscus fuscus |
| Metadata16 | Individual information | age | list | Enter exact age (years) if this is known (and it is not a pullus). 'pullus', 'subadult' or 'adult\_unknown' - otherwise |
| Metadata16 | Individual information | sex | list | [male or female] 'unknown' otherwise |
| Metadata16 | Individual information | sexing\_method | string | enter method for sexing e.g. 'dna', 'morphology', 'behaviour', 'none\_yet' |
| Metadata16 | Morphometrics | weight | number | [grams] |
| Metadata16 | Morphometrics | scull | number | [millimeters], head plus bill |
| Metadata16 | Morphometrics | tarsus | number | [millimeters] |
| Metadata16 | Morphometrics | wing | number | [millimeters] from carpal joint to tip of primary, flattened wing |
| Metadata16 | Breeding status | breeding\_stage | list | Breeding stage at handling, pick from drop-dowm list |
| Metadata16 | Breeding status | eggs | number | number of eggs at handling |
| Metadata16 | Breeding status | chicks | number | number of chicks at handling |
| Metadata16 | Breeding status | hatching\_success | Binomial | '1' -if at least one egg hatched and a chick /was produced otherwise '0' or 'unknown' |
| Metadata16 | Breeding status | breeding\_success | Binomial | 1' if at least one chick fledged or survived to a 'large chick' stage (specify in next collumn) Otherwise '0' or 'unknown' |
| Metadata16 | Breeding status | breeding\_success\_criterion | list | Breeding success should be one if at least one chick has survived to fledging or at least to a "large chick" stage. "Large chick stage" can be defined in several ways (see list below) and this field specifies which criterion has been used '10d' -if at least one chick survived ≥10 days '15d' -if at least one chick survived ≥15 days, '20d' -if at least one chick survived ≥20 days, '25d' -if at least one chick survived ≥25 days, '30d' -if at least one chick survived ≥30 days, 'fledging' -if at least one chick survived to fledging, 'none' -if breeding success is unknown |
| Metadata16 | Breeding location | country | list | name of country, e.g. norway, russia, iceland, faroe islands, great britain |
| Metadata16 | Breeding location | colony | list | name of colony or location - Choose the lowest common geographical unit e.g name of cliff or island if the work has been done at one location only, name of location/area if the work has been at several locations |
| Metadata16 | Breeding location | colony\_latitude | number | Decimal degrees e.g. 65.4967 |
| Metadata16 | Breeding location | colony\_longitude | number | Decimal degrees e.g. 15.4967, use positive values for East and negative values for West |
| Metadata16 | Breeding location | nest\_id | string | nest identification |
| Metadata16 | Samples | blood\_sample | list | select the appropriate option, i.e.If and for what purposes blood was collected |
| Metadata16 | Samples | feather\_sample | list | select the appropriate option, i.e.If and for what purposes feathers were collected |
| Metadata16 | Samples | other\_samples | string | Requested if appropriate. If any other samples are obtained from an individual carrying a GLS logger please specify what type and for what purposes. (e.g. “cloacal swabs, immunological studies”, “Ticks or other parasites sampled”, etc.) |
| Metadata16 | Other | data\_responsible | string | Name of responsible person, who should be contacted regarding publication of data e.g 'H Strom', 'S Descamps',etc |
| Metadata16 | Other | back\_on\_nest | list | 'yes' -if bird returned back on nest after handling 'no' or 'unknown' - otherwise |
| Metadata16 | Other | logger\_mount\_method | string | Enter where the logger was mounted to the bird |
| Metadata16 | Other | comment | string |  |
| Metadata16 | Other | other relevant variables, e.g. 'gonys', 'culmen, | string | Enter additional variables if relevant e.g 'gonys', 'culmen' |
| Currentdatabase | Nonspecified | 'date\_time' | time | date and time of the calculated position (DD.MM.YYYY HH:MM) |
| Currentdatabase | Nonspecified | logger' | string | logger ID and logger model ( e.g N032\_c65) OBS! See 'logger\_yeartracked' below!! |
| Currentdatabase | Nonspecified | logger\_id' | string | specific ID for each GLS logger (e.g N032). OBS! See 'logger\_yeartracked' below!! |
| Currentdatabase | Nonspecified | logger\_model' | list | GLS logger model (e.g c65, mk4083 etc) |
| Currentdatabase | Nonspecified | 'year\_tracked' | list | 'year\_tracked' specify the tracking period from one breeding season to the next (e.g. '2013\_14'). If the logger contain more than one year of data , there will be more than one year tracked (e.g. '2013\_14 for the first year and '2014\_15' for the second year tracked. |
| Currentdatabase | Nonspecified | 'year\_deployed' | list | year logger was deployed (YYYY) |
| Currentdatabase | Nonspecified | 'year\_retrieved' | list | year logger was retrieved (YYYY) |
| Currentdatabase | Nonspecified | 'ring\_number' | string | id-number on the metal ring (e.g CA443314) |
| Currentdatabase | Nonspecified | 'euring\_code' | list | what bird ringing scheme the metal ring belongs to (e.g. 'NOS' for Norway/Stavanger museum) |
| Currentdatabase | Nonspecified | 'species' | list | english name of the species (e.g 'Common guillemot') |
| Currentdatabase | Nonspecified | 'colony' | list | name of the colony the logger was deployed and retrieved in (e.g 'Isle of May') |
| Currentdatabase | Nonspecified | 'lon\_raw' | number | longitude calculated from time (GMT) of noon or midnight and date. |
| Currentdatabase | Nonspecified | 'lat\_raw' | number | latitude calculated from the length of day or night and date. |
| Currentdatabase | Nonspecified | 'lon\_smooth1' | number | averaging consequtive noon-midnight or midnight-noon longitudes from 'lon\_raw'. |
| Currentdatabase | Nonspecified | 'lat\_smooth1' | number | averaging consequtive noon-midnight or midnight-noon latitudes from 'lat\_raw'. |
| Currentdatabase | Nonspecified | lon\_smooth2' | number | two point moving average over the lon\_smooth1-lat\_smooth1 coordinates using spherical trigonometry. We adviced to use these double smoothed positions for plotting and interpretations. |
| Currentdatabase | Nonspecified | lat\_smooth2' | number | two point moving average over the lon\_smooth1-lat\_smooth1 coordinates using spherical trigonometry. We adviced to use these double smoothed positions for plotting and interpretations. |
| Currentdatabase | Nonspecified | 'disttocol\_s2' | number | distances from double smoothed positions (lon\_smooth2, lat\_smooth2) to the breeding colony calculated as great circle distances assuming a spherical earth. |
| Currentdatabase | Nonspecified | 'eqfilter1' | Binomial | this filter identifies fixed periods where the latitudes are assumed to be unreliable during equinox (1 = reliable/outside fixed equinox period, 0 = unreliable/inside equinox). Example, value 0 can represent the periods from 10 Sep to 18 Oct and 20 Feb to 5 Apr. |
| Currentdatabase | Nonspecified | 'eqfilter2' | Binomial | this is a filter set by the analyzer rough visual inspection of positions on a map, and identifies the periods in which the latitudes for indivual loggers are clearly affected by the equinoxes (1 reliable, 0 unreliable). This filter is mainly used as a supplement to eqfilter1 (see eqfilter3). |
| Currentdatabase | Nonspecified | 'eqfilter3' | Binomial | this filter combines the information from 'eqfilter1' and 'eqfilter2'. eqfilter3 is 1 when efilter1 is 1 AND eqfilter2 is 1. This filter is the most conservative filter of the three eqfilters. We recomended to use this one. |
| Currentdatabase | Nonspecified | 'lat\_smooth2\_eqfilt3' | number | contain data from 'lat\_smooth2', when 'eqfilter3'is 1. |
| Currentdatabase | Nonspecified | 'sex' | list | sex of the bird |
| Currentdatabase | Nonspecified | 'morph' | string | morph of the bird if morph has been assigned. Usually only applied for 'briddled' and 'non\_briddled' common guillemonts. |
| Currentdatabase | Nonspecified | 'subspecies' | list | subspecies of the bird if assigned |
| Currentdatabase | Nonspecified | 'age' | list | identifies the bird as adult or juvenile, or the specific age of the bird if known. |
| Currentdatabase | Nonspecified | 'col\_lon' | number | longitude of the colony. |
| Currentdatabase | Nonspecified | 'col\_lat' | number | latitude of the colony. |
| Currentdatabase | Nonspecified | 'tfirst' | Time | time of first sun event (DD.MM.YYYY HH:MM) for a pair of sun events (sunrise-sunset or sunset-sunrise) used for calculating positions |
| Currentdatabase | Nonspecified | 'tsecond' | Time | time of second sun event (DD.MM.YYYY HH:MM) for a pair of sun events (sunrise-sunset or sunset-sunrise) used for calculating positions |
| Currentdatabase | Nonspecified | 'twl\_type' | Binomial | identify whether 'tfirst' is sunrise (1) or sunset (2) |
| Currentdatabase | Nonspecified | 'conf' | list | confidence sun event (time when light cross the selected treshold value) set by analyzer (9 is high confidence, 4 is high confidence but adjusted manually by analyzer, 3 is low confidence and adjusted manually by analyzer, 2 is low confidence |
| Currentdatabase | Nonspecified | 'sun' | number | the sun elevation angle used to calculated the positions from the assigned sun events. Negative values refers to degrees below horizon. |
| Currentdatabase | Nonspecified | 'software' | list | software used to inspect the light data and assign the sun events. Biotrack (for loggers produced by Biotrack) or IntiProc (loggers produced by Migrate technology) |
| Currentdatabase | Nonspecified | 'light\_threshold' | number | selected treshold that defines a sunset/sunrise |
| Currentdatabase | Nonspecified | 'analyzer' | string | name of person inspecting the light data and setting the sun events |
| Currentdatabase | Nonspecified | 'data\_responsible' | string | name of data responsible, person to contact about the loggerdata. Usually project/field leader at a given field location. |
| Currentdatabase | Nonspecified | 'logger\_yeartracked' | string | 'logger\_id' +'logger\_model' + 'year\_tracked'(e.g N032\_c65\_2014\_15) - this value will always be unique for each track. In multiyear/multicolony studies, loggers may be reused on different birds, and the same 'logger\_id' set by the producers can have been set to two different loggers by chance |
| Currentdatabase | Nonspecified | 'posdata\_file' | string | identifies which file the trackingdata is obtained from (e.g 'posdata\_FRAARC\_anda\_2015') |
| Processingfile | Nonspecified | logger\_id\_retrieved | string | specific ID for each GLS logger (e.g N032). OBS! See 'logger\_yeartracked' below!! |
| Processingfile | Nonspecified | logger\_model\_retrieved | list | GLS logger model (e.g c65, mk4083 etc) |
| Processingfile | Nonspecified | year\_retrieved | list | year logger was retrieved (YYYY) |
| Processingfile | Nonspecified | year\_tracked | list | 'year\_tracked' specify the tracking period from one breeding season to the next (e.g. '2013\_14'). If the logger contain more than one year of data , there will be more than one year tracked (e.g. '2013\_14 for the first year and '2014\_15' for the second year tracked. |
| Processingfile | Nonspecified | logger\_producer | list | Name of producer [migratetech;biotrack;staroddi;...] |
| Processingfile | Nonspecified | ring\_number | string | id-number on the metal ring (e.g CA443314) |
| Processingfile | Nonspecified | euring\_code | list | what bird ringing scheme the metal ring belongs to (e.g. 'NOS' for Norway/Stavanger museum) |
| Processingfile | Nonspecified | species | list | english name of the species (e.g 'Common guillemot') |
| Processingfile | Nonspecified | sex | list | sex of the bird |
| Processingfile | Nonspecified | morph | string | morph of the bird if morph has been assigned. Usually only applied for 'briddled' and 'non\_briddled' common guillemonts. |
| Processingfile | Nonspecified | subspecies | list | subspecies of the bird if assigned |
| Processingfile | Nonspecified | age | list | identifies the bird as adult or juvenile, or the specific age of the bird if known. |
| Processingfile | Nonspecified | colony | list | name of the colony the logger was deployed and retrieved in (e.g 'Isle of May') |
| Processingfile | Nonspecified | col\_lat | number | latitude of the colony. |
| Processingfile | Nonspecified | col\_lon | number | longitude of the colony. |
| Processingfile | Nonspecified | year\_deployed | list | year logger was deployed (YYYY) |
| Processingfile | Nonspecified | date\_deployed | Date | [dd.mm.yyyy] when logger was deployed |
| Processingfile | Nonspecified | date\_retrieved | Date | [dd.mm.yyyy] when logger was retrieved |
| Processingfile | Nonspecified | firstdate\_light | Date | [dd.mm.yyyy] When logger first recorded a valid twiglight event for a given logger year |
| Processingfile | Nonspecified | lastdate\_light | Date | [dd.mm.yyyy] When logger last recorded a valid twiglight event for a given logger year |
| Processingfile | Nonspecified | first\_aut\_eq | Date | [dd.mm.yyyy] Start of autumn equinox effect |
| Processingfile | Nonspecified | last\_aut\_eq | Date | [dd.mm.yyyy] end of autumn equinox effect |
| Processingfile | Nonspecified | first\_spring\_eq | Date | [dd.mm.yyyy] Start of spring equinox effect |
| Processingfile | Nonspecified | last\_spring\_eq | Date | [dd.mm.yyyy] end of spring equinox effect |
| Processingfile | Nonspecified | software | list | software used for analysis |
| Processingfile | Nonspecified | light\_threshold | number | light\_threshold (lux) used to define a sunevent |
| Processingfile | Nonspecified | trn\_file | string | filename of trn.file, LoggerID(reconstructed or not?)\_YearRetrieved\_LoggerModel |
| Processingfile | Nonspecified | processing\_file | string | filename of processing table, one per population per year |
| Processingfile | Nonspecified | posdata\_file | string | filename of positional table, one per population per year |
| Processingfile | Nonspecified | logger\_download\_success | list | Was logger successfully downloaded |
| Processingfile | Nonspecified | logger\_date\_failed | Date | date when logger failed |
| Processingfile | Nonspecified | analyzer | string | name of data responsible, person who processed data |
| Processingfile | Nonspecified | data\_responsible | string | name of data responsible, person to contact about the loggerdata. Usually project/field leader at a given field location. |
| Processingfile | Nonspecified | sun1\_db | number | Sunangle set for a given loggeryear rounded up |
| Processingfile | Nonspecified | sun2\_db | number | Sunangle set for a given loggeryear |
| Processingfile | Nonspecified | comment1 | string |  |
| Processingfile | Nonspecified | nest\_id\_deployed | string | nest identification |
| Processingfile | Nonspecified | nest\_id\_retrieved | string | nest identification |
| Processingfile | Nonspecified | bm\_deployed | Unknown |  |
| Processingfile | Nonspecified | bm\_retrieved | Unknown |  |
| Processingfile | Nonspecified | winter\_area\_code | list |  |
| Processingfile | Nonspecified | winter\_area | list |  |
| Processingfile | Nonspecified | sun\_prelim | Unknown |  |
| Processingfile | Nonspecified | comment\_sun | string |  |
| Processingfile | Nonspecified | start autumn migration | Date |  |
| Processingfile | Nonspecified | arrival winter area | Date |  |
| Processingfile | Nonspecified | start spring migration | Date |  |
| Processingfile | Nonspecified | arrival breeding location | Date |  |
| Processingfile | Nonspecified | comment\_migration dates | string |  |
| Processingfile | Nonspecified | first egglay | Date |  |
| Processingfile | Nonspecified | comment egglay | string |  |
| Processingfile | Nonspecified | delete dates | Unknown |  |
| Processingfile | Nonspecified | estimated\_autumn\_departure | Date |  |
| Processingfile | Nonspecified | estimated\_spring\_arrival | Date |  |
| Processingfile | Nonspecified | duration autumn migration | number |  |
| Processingfile | Nonspecified | WINTER | Unknown |  |
| Processingfile | Nonspecified | duration\_spring\_,migration | number |  |
| Processingfile | Nonspecified | year\_tracked | list |  |
| Processingfile | Nonspecified | InHabitat N | Unknown |  |
| Processingfile | Nonspecified | InHabitat S | Unknown |  |
| Processingfile | Nonspecified | HillEkstrom | Unknown |  |
| Processingfile | Nonspecified | comment\_HE og INHabitat | string |  |
| Processingfile | Nonspecified | lat vs time plots | Unknown |  |
| Processingfile | Nonspecified | Loess\_filter k | Unknown |  |
| Processingfile | Nonspecified | h\_norw(winter) | Unknown |  |
| Processingfile | Nonspecified | h\_svalbard | Unknown |  |
| Processingfile | Nonspecified | lagt inn i Eider positions 2009\_2013 | Unknown |  |
| Processingfile | Nonspecified | deployed\_retrieved | Binomial | Quality check |
| Processingfile | fokus på år | last-first | Binomial | Quality check |
| Processingfile | fokus på år | first\_aut\_eq | Binomial | Quality check |
| Processingfile | fokus på år | last\_aut\_eq | Binomial | Quality check |
| Processingfile | fokus på år | first\_spring\_eq | Binomial | Quality check |
| Processingfile | fokus på år | last\_spring\_eq | Binomial | Quality check |
| Processingfile | fokus på måned | first\_aut\_eq | Binomial | Quality check |
| Processingfile | fokus på måned | last\_aut\_eq | Binomial | Quality check |
| Processingfile | fokus på måned | first\_spring\_eq | Binomial | Quality check |
| Processingfile | fokus på måned | last\_spring\_eq | Binomial | Quality check |
| Processingfile | kopi av dato-kollonnene for laging av pivot-tabeller | year\_aut\_eq | list |  |
| Processingfile | kopi av dato-kollonnene for laging av pivot-tabeller | date\_retrieved | Date |  |
| Processingfile | kopi av dato-kollonnene for laging av pivot-tabeller | firstdate\_light | Date |  |
| Processingfile | kopi av dato-kollonnene for laging av pivot-tabeller | lastdate\_light | Date |  |
| Processingfile | kopi av dato-kollonnene for laging av pivot-tabeller | first\_aut\_eq | Date |  |
| Processingfile | kopi av dato-kollonnene for laging av pivot-tabeller | last\_aut\_eq | Date |  |
| Processingfile | kopi av dato-kollonnene for laging av pivot-tabeller | first\_spring\_eq | Date |  |

1. Data exist from other types of loggers from different producers and in the long run this will need to be taken into consideration, especially so if the project expands to incorporate data from other projects. The majority of the loggers we include today and all loggers bought by SEATRACK are however either Biotrack or Migratetech loggers of the makes described above. [↑](#footnote-ref-1)