Database documentation: obs_lfs

B.M. Sanders & K. A. Mackay

NIWA Fisheries Data Management
Database Documentation Series

Revised January 2008

Contents

List of Figures

Revision History

Version	Change	Date	Responsible
1.1	MAF Internal report No. 199	1993	K. A. Mackay
1.2	NIWA Internal report No. 42	1999	K. A. Mackay & B. M. Sanders
1.3	NIWA Internal report No. 95	2000	K. A. Mackay & B. M. Sanders
2.1	Observer **squid** database incorporated into **obs_lfs**. Business rules added.	November 2001	Brian Sanders
2.1.1	Appendix 1- Reference code table for 'sample weight method', code 4 added.	19 June 2003	Brian Sanders
2.1.2	Added attributes age and age_actual to t_nonfish_catch.	14 June 2004	Brian Sanders
2.1.3	'Should' definition added, section 7.1	10 November 2005	Brian Sanders
3.1.0	t_purseseine table added for observer purse seine trips.	January 2006	Brian Sanders
3.1.1	Added attribute company to t_trip_master for purseseine.	January 2006	Brian Sanders
3.1.2	t_line_catch modified to store purseseine catch; hold_no field added and discard renamed state. Renamed t_catch.	January 2006	Brian Sanders
4.1	Seabird warp_strike observations (trawl) tables added.	March 2006	Brian Sanders
4.1.2	Added v_station view with position as decimal (to 0.1 deg) Modified v_line and v_tow views, position as decimal (to 0.1 deg)	07 July 2006	Brian Sanders
4.1.3	Referred to l_line in section 2. Corrected nonfish station table name in business rules section, & no_fl business rule	14 February 2007	David Fisher
4.1.4	.Added male gonad stages to t_lth	16 January 2008	Brian Sanders

1 Database Document Series

The National Institute of Water and Atmospheric Research (NIWA) currently carries out the role of Data Manager and Custodian for the fisheries research data owned by the Ministry of Fisheries.

The Ministry of Fisheries data set incorporates historic research data, data collected more recently by MAF Fisheries prior to the split in 1995 of Policy to the Ministry of Fisheries and research to NIWA, and currently data collected by NIWA and other agencies for the Ministry of Fisheries.

This document is a brief introduction to the observer length frequency database **obs_lfs**, and is a part of the database documentation series produced by NIWA. It supersedes the previous documentation by Sanders & Mackay (2000)[1] on this database and the documentation on the **squid** database by Annette Atkins (1993)[2]. All documents in this series include an introduction to the database design, a description of the main data structures accompanied by an Entity Relationship Diagram (ERD), and a listing of all the main tables. The ERD graphically shows the relationships between the tables in **obs_lfs**, and the relationships between these tables and other databases.

This document is intended as a guide for users and administrators of the **obs_lfs** database.

Access to this database and data are restricted to specific Nominated Personnel as specified in the current Schedule 6 of the Data Management contract between the Ministry of Fisheries and NIWA. Any requests for data should in the first instance be directed to the Ministry of Fisheries.

2 Observer Length Frequency Database

The **obs_lfs** database is one of two databases dedicated to information collected by the Scientific Observer Programme (SOP). The first is the **obs** database, which contains catch and effort information for observed vessels. The second is the **obs_lfs,** which contains length frequency and biological data for commercial species as measured by the observers, as well as relevant trip and tow information.

This database is the major source of length frequency data from commercial fishing operations, and so plays a major role in the stock assessment process. Currently the **obs_lfs** database holds information for the major species including hoki, orange roughy, southern blue whiting, oreos, jack mackerels, blue mackerel, bluenose, hake, barracouta, bass, ling, warehou, stargazers, spiny dogfish, scampi, gemfish and squid. Smaller amounts of data are available for species such as alfonsino, gurnard, john dory, school shark, snapper, tarakihi, ribaldo, hapuku, red cod, and cardinal fish. An increasing amount of Antarctic toothfish and Patagonian toothfish data are also being held within the **obs_lfs** database.

[1] Sanders, B.M. & Mackay, K.A. 2000 Database documentation. obs_lfs. *NIWA Internal Report No. 95*. 26p.
[2] Atkins, A. 1993. Marine Research database documentation. 5. squid. *MAF Fisheries Greta Point Internal Report No. 204*, 17 p.

The New Zealand arrow squid data collected by scientific observers on both squid trawlers and jiggers, previously held in the **squid** database, have been incorporated into the **obs_lfs** database. The biological data consists mostly of southern arrow squid, *Nototodarus sloanii,* and a lesser amount of *N. gouldi*.

A non-fish bycatch data collection form for Scientific Observers was introduced in 1994, replacing the "Observer Seal Sample Data Sheet". Data recorded on the non-fish bycatch form have been incorporated within the **obs_lfs** database. Species that have been recorded and entered into the database at the time of writing include bottlenose dolphin, common dolphin, dusky dolphin, New Zealand fur seal, Hooker's sea lion, leopard seal and a range of seabirds. Data from the "seal" form were transferred from the table *sealtable* in the **obs** database, into the **obs_lfs** database, (covering trips from 541 to 779).

Longline vessel data collected by the Scientific Observer Programme, beginning in 1993 with the Kermadec Fishery Management Area exploratory research programme, have also been incorporated into the **obs_lfs** database. The longline data are now expanded to include ling and toothfish trips.

Surface longline data collected by the Scientific Observer Programme is contained in a separate database **l_line**.

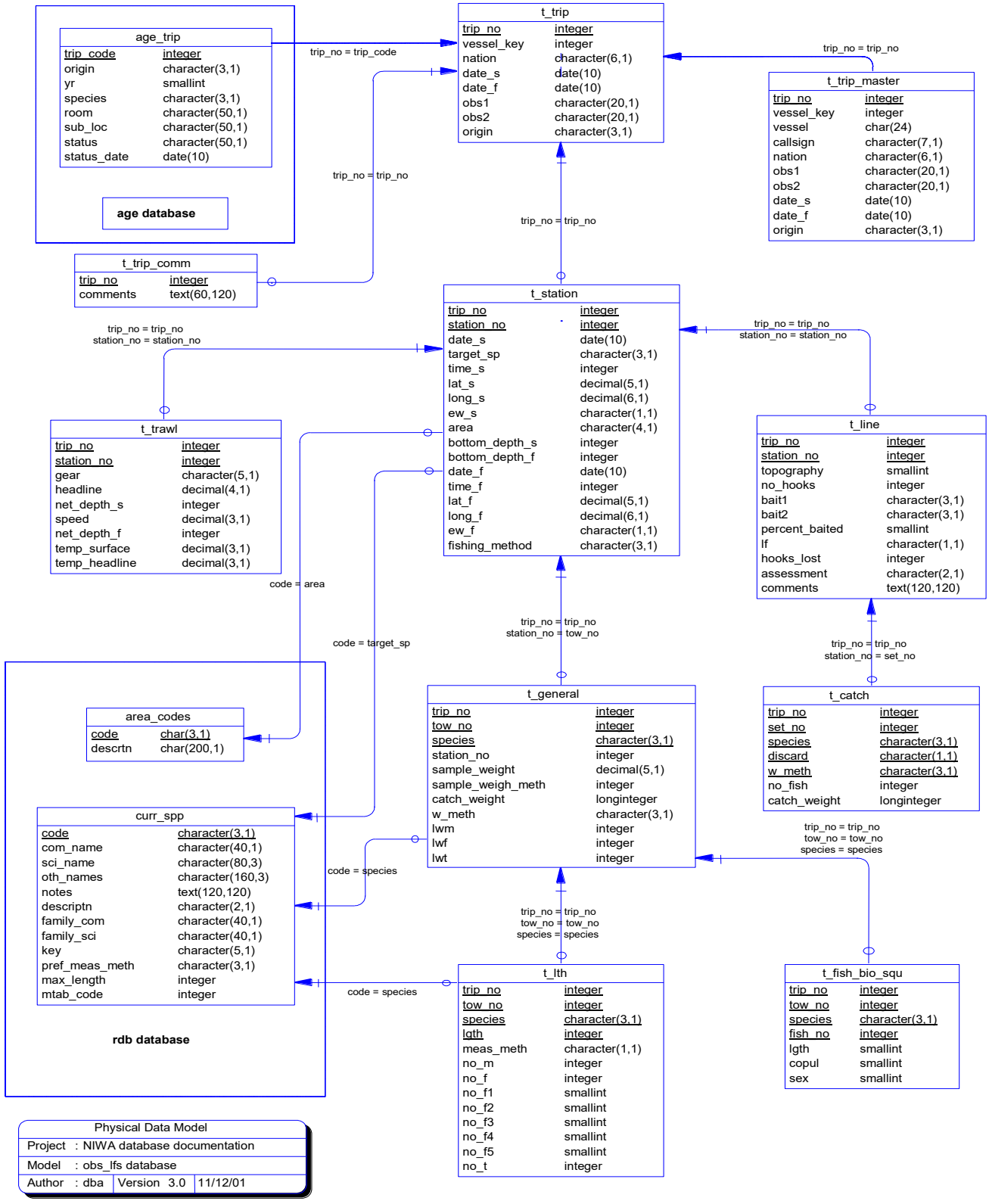


Figure 1: Entity Relationship Diagram (ERD) of the obs_lfs database. The table *t_trip_master* is restricted to the database administrator and only the view *t_trip* is visible to users of the database.

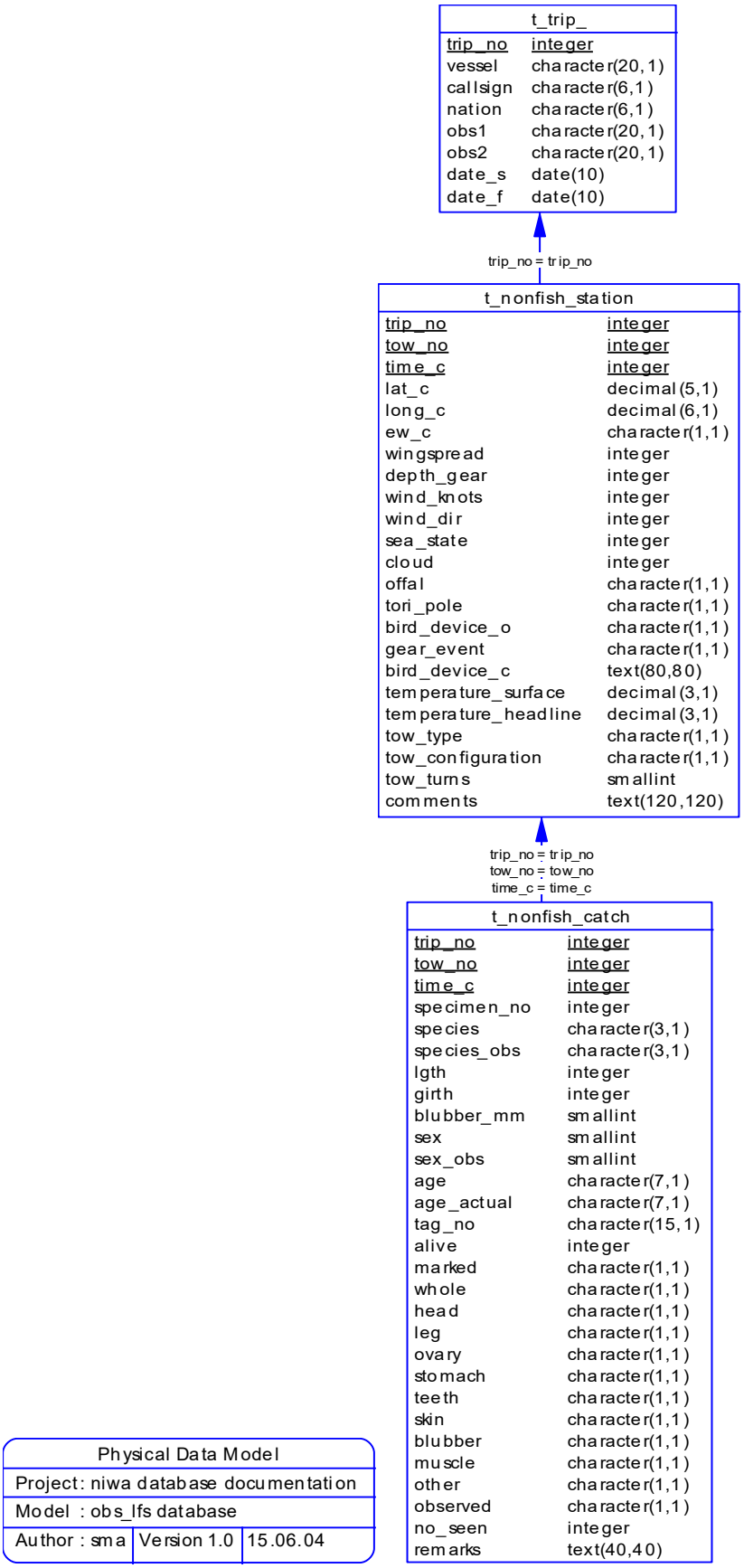


Figure 2: ERD showing the relationship between the non-fish bycatch tables in the **obs_lfs** database.

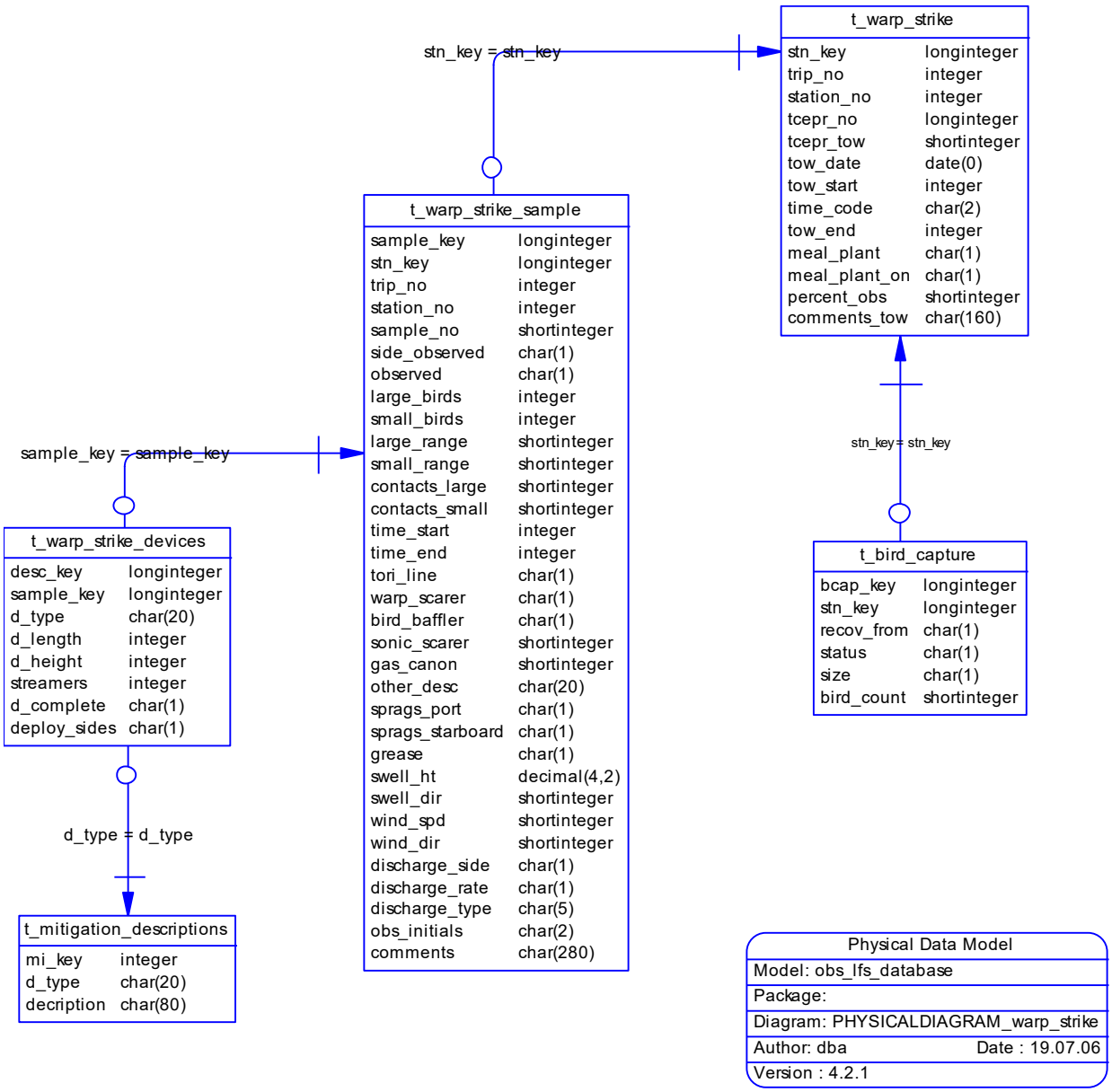


Figure 3: ERD showing the relationship between the seabird warp-strike tables in the obs_lfs database.

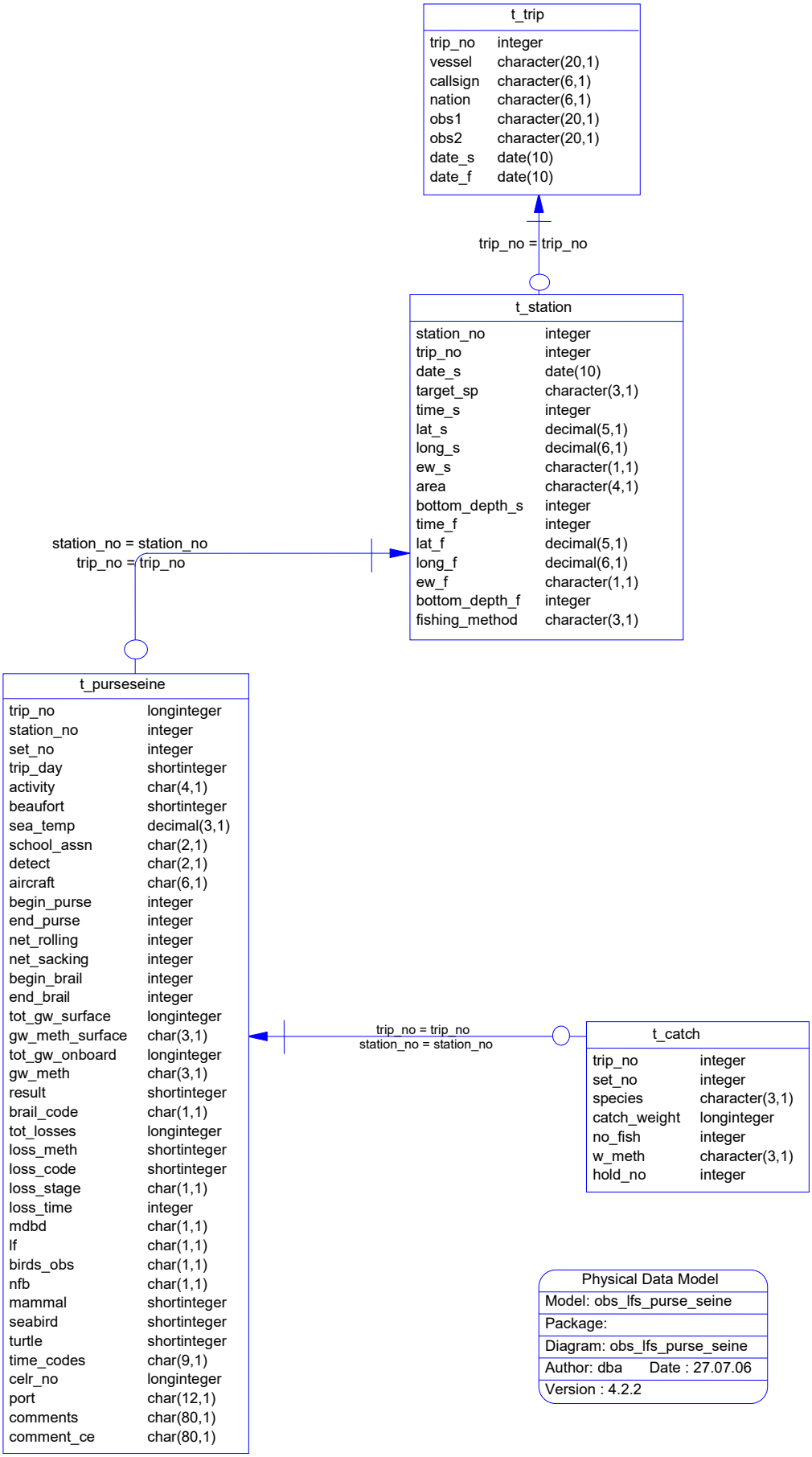


Figure 4 ERD showing the relationship between the observer purse-seine tables in the obs_lfs database.

3 Data Structures

3.1 Table relationships

The **obs_lfs** database comprises various related tables. The ERD (Figures 1 and 2) shows the logical structure of the database and its entities (each entity is implemented as a database table) and relationships between these tables and tables in other databases. All the tables' attributes are shown in the ERD. The primary keys[2] are underlined for each table. The primary key has a unique index attached to it, that are generally listed using the format:

Indices: UNIQUE index_name ON (*attribute* [*, attributes*])

where the attribute(s) make up the primary key (the key attributes) and the index name is the primary key name. Note that the typographical convention for the above format is that square brackets [] may contain more than one item or none at all. A unique index prevents records with duplicate key values from being inserted into the table; e.g., a new trip with an existing trip number.

The **obs_lfs** database is implemented as a relational database. That is, each table is a special case of the mathematical construct known as a *relation* and hence elementary relation theory is used to deal with the data within tables and the relationships between them. All relationships in **obs_lfs** are of the type *one-to-many*[3]. This is shown in the ERD by connecting a single line (indicating 'many') from the child table; e.g., *t_station*, to the parent table; e.g., *t_trip*, with an arrowhead (indicating 'one') pointing to the parent. For example, consider the relationship between the tables, *t_trip* (the parent table) and *t_station* (the child table). Any one observer trip in *t_trip* can have one or more stations in *t_station*, but any one station can only be a part of one observer trip. Note that the word 'many' applies to the possible number of records another is associated with. For a given instance, there might be zero, one, two, or more associated records, but if it is ever possible to have more than one, we use the word 'many' to describe the association.

Note that the one-to-many relationships can be either mandatory or optional. The optional relationship, denoted in the ERD by the symbol "O" at one or both ends of the relationship line, means that a record does not have to have any associated records. Conversely, the mandatory relationship denoted in the ERD by a bar symbol across the relationship line, means that a record has to have at least one associated record. For example, if we consider again the one-to-many relationship between the tables *t_trip* and *t_station*, which has a mandatory 'one' and an optional 'many'. This means that one trip record can have zero or more (many) stations within it, but one station must have one, and only one, associated record in the trip table.

[2] The primary key is an attribute or combination of attributes whose values are unique for that record.
[3] A one-to-many relationship is where one record (the *parent*) in a table relates to one or many records (the *child*) in another table; e.g., one trip in *t_trip* can have many stations in *t_station* but any one stratum can only come from one survey.

These relationships are enforced in the database by the use of referential constraints[4]. Constraints do not allow orphans to exist in any table; i.e., where a child record exists without a related parent record. This may happen when:

i. a parent record is deleted;
ii. the parent record is altered so the relationship is lost;
iii. or a child record is entered without a parent record.

All constraints in **obs_lfs** prevent these from occurring. Constraints are shown in the table listings by the following format:

```
Referential:      error message (attribute[, attribute]) |INSERT|
                                                         |DELETE|
                  parent table (attribute[, attribute])
```

Items stacked between vertical lines | | are options of which one must be chosen.

In the Empress RDBMS a constraint can either be INSERT (prevents (ii) and (iii) from occurring) or DELETE (prevents (i) or (ii) from occurring). For example, consider the following constraint found in the table *t_station*:

```
Referential:      Invalid trip (survey) INSERT t_trip (trip_no)
```

This means that the value of the attribute *trip_no* (that is, one trip) in the current station record must already exist in the parent table *t_trip* or the record will be rejected and the following message will be displayed:

```
*** User Error: insert constraint 'Invalid trip' violation
```

All tables in this database are indexed. That is, attributes that are most likely to be used as a searching key have like values linked together to speed up searches. These indices are listed using the following format:

```
Indices:    NORMAL (2, 15) index_name ON (attribute [, attributes ])
```

Note that indices may be simple, pointing to one attribute, or composite pointing to more than one attribute. The numbers "…(2, 15)…" in the syntax are Empress DBMS default values relating to the amount of space allocated to index storage.

3.2 Database Design

The top-level table in this database is *t_trip* (Table 1), which contains records for each trip on which length frequency data were collected. Each trip record has a unique attribute *trip_no*, which is the primary key for this table. This primary key also provides a one-to-many relationship to the **age** database. The attribute *trip_no* in *t_trip* can be used to access age information for all the fish sampled from that trip, held in the **age** database.

Each trip can either have many tows or sets from which fish were sampled, linking *t_trip* to *t_station* with a one-to-many relationship (Table 2). Each tow/set has an attribute *station_no*, which with *trip_no*, forms the primary key for this table. The attribute *area* lists Fisheries Management Area (FMA) and research area codes, and is a foreign key to the table *area_codes* in the **rdb** database. There is a referential constraint to the **rdb** database to make sure that only valid codes are inserted into this table.

[4] Also known as integrity checks.

Each species sampled from a tow or set produces a record in the table *t_general* (Table 7), which contains weights for the sample and the catch.

Length frequency and gonad staging (female fish only up to 2006/2007 fishing year) records are held in the table *t_lth* (Table 8). Previously the gonad stages were stored in a *t_gonad_stage* table. Both these tables had the same primary key, and for the database to be in the third normal form, they have been amalgamated. The foreign key *meas_meth* is contained in each record of *t_lth*. This provides a link through a referential constraint to the table *t_fish_meas_codes* in the **rdb** database.

Contained in this database are a number of views of the table *t_general* (Views 3 - 8). Each view is a 'filter' in to this table, which only reveals records for a selected species.

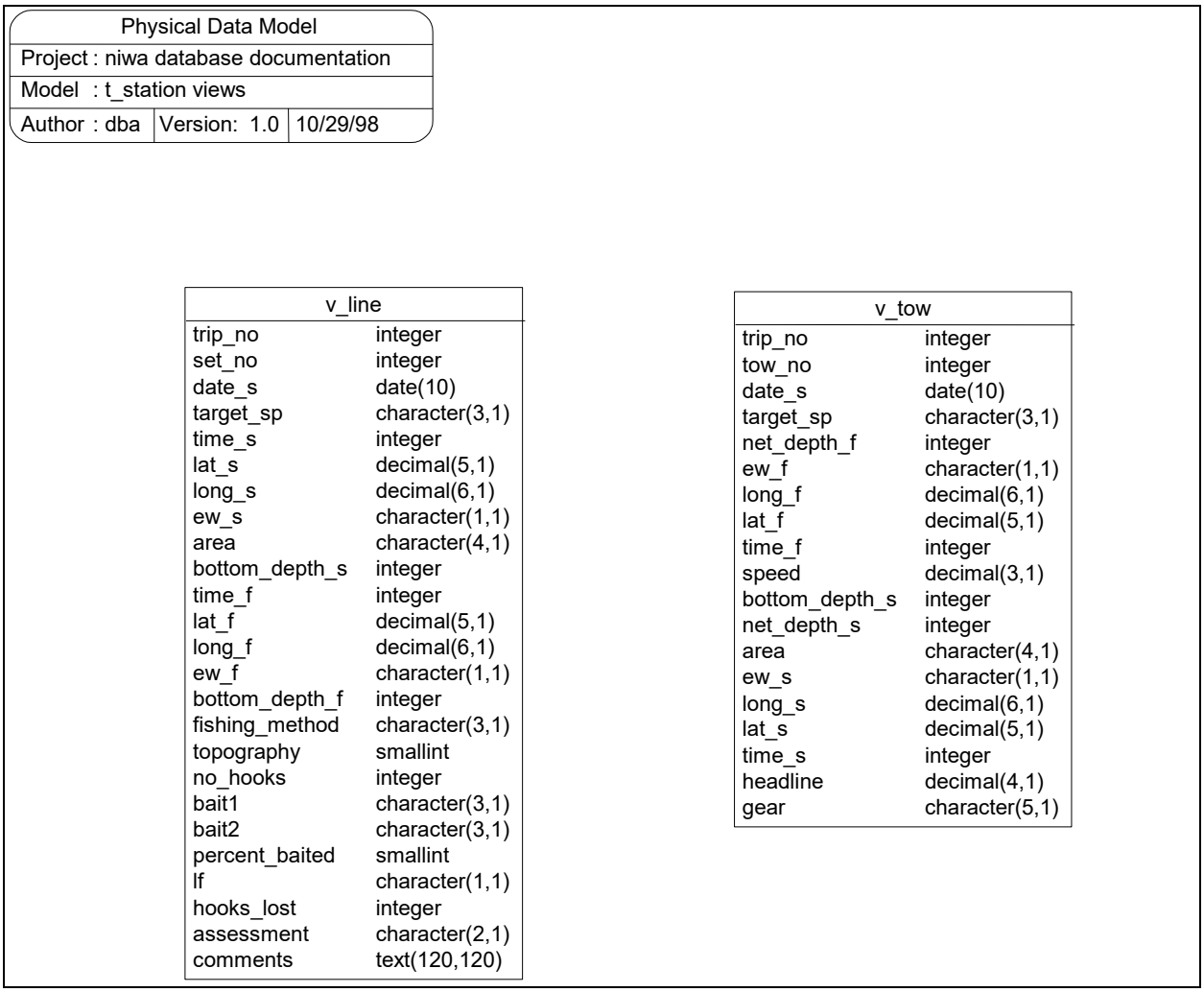


Figure 4: Showing several main views for trawl and longline station information in the **obs_lfs** database.

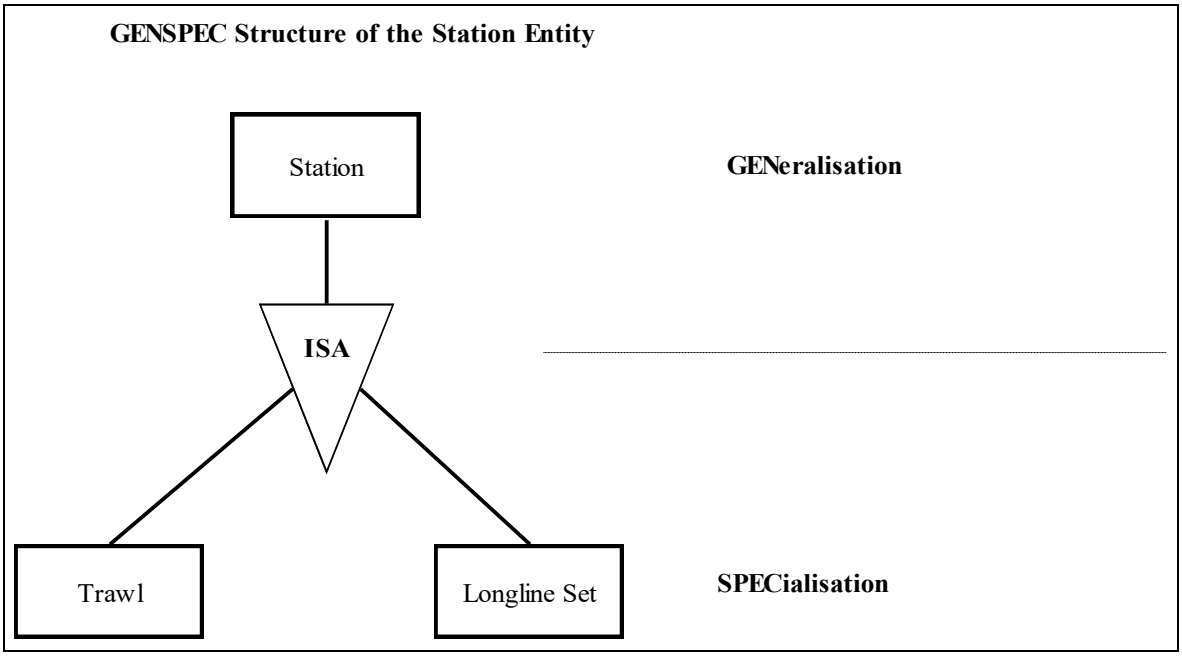


Figure 5: GENSPEC diagram for Trawl and Longline station data.

3.2 Generalised station data

The inclusion of observed longline trips in addition to trawl trips into the **obs_lfs** database, has required modifying the database to store station data from several sources. A powerful abstraction called generalisation, that allows objects of different types to be considered as examples of a higher-level set, has been employed for this purpose; e.g., a trawl and a longline set, are seen to be examples of a station. This can be represented by the GENSPEC structure (GENeralisation / SPEcialisation) seen in Figure 5.

The generalisation and specialisation are pictured using a triangle containing the words “IS A” to connect the components to each other and to the higher-level entity. The generalized higher-level entity, implemented as the table *t_station* (Table 3) contains the common attributes of all examples of a station; e.g., date, start and finish time, latitude, longitude, etc. The specialised entities, implemented as the tables *t_trawl* (Table 4) and *t_set* (Table 5) contain only attributes relevant to their specific types. For example; headline height is stored in *t_trawl*, number of hooks is stored in *t_set*. The attributes of the higher-level entity are "inherited" by the lower-level ones. Specifically, this is achieved by views, which join the higher- and lower-level entities together. These views for trawl (*v_tow*) and longline (*v_line*) data, are detailed in **obs_lfs** views, see section 6.1 and 6.2 of this report.

Note, that *v_tow* has compatible attributes with the *t_tow* table that existed prior to the addition of longline data and the need for the current GENSPEC structure. (The *t_tow* table was documented in the previous documentation by Mackay (1993) on this database.)

3.3 Trawl

Trawl caught fish make up the bulk of length frequency data stored in the **obs_lfs** database. Data collected by various agencies are now held in the **obs_lfs** database, as recorded in the attribute origin in the *t_trip* table. Current origin codes are listed in Appendix 1. As mentioned, scientific observers on board trawl vessels collect information on catch and effort, which is recorded in logbooks. SOP logbook data are subsequently stored in detail in the **obs** database. For the tows where length frequency samples have been taken by scientific observers a sub-set

of relevant station data are stored in **obs_lfs**. For each species sampled, green weight and method of weighting are extracted from the **obs** database, and stored in the *t_general* table of **obs_lfs**, along with the sample weights and their method codes recorded on the length frequency forms.

Trawl data sets collected from sources other than the SOP do not have logbook data stored in the **obs** database. Only the relevant details as required for sampled tows, are held in the **obs_lfs** database.

Industry collected data includes the Trawl, Catch, Effort and Processing Return (TCEPR) number and then the shot number per TCEPR form for each trip. To retain compatibility with the existing data structure, industry sampled tows have been assigned a station number sequentially from 1 for each trip, as it was not possible to derive the actual tow number from the data. The TCEPR and 'shot number' along with assigned station number by trip are stored in a reference table *t_tcpern*, accessible by the database administrator.

3.4 Longline

The SOP longline trips do not have catch and effort logbook data stored in the **obs** database. Therefore all the set and catch details recorded on the set form are stored directly into the **obs_lfs** database, unlike trawling trips, where only a sub-set of trawl station data with length frequency samples is stored in **obs_lfs**. For longline trips, each set is stored in *v_line*, with the catch for each set (if any) stored in *t_catch*. For each set, each species that has been sampled will have a *t_general* record and one or more *t_lth* records (the same as for trawl caught length frequency samples).

3.5 Non-fish bycatch

Data recorded on the non-fish bycatch form are stored in two tables, *t_nonfish_station* and *t_nonfish_catch* (Figure 2).

On the non-fish bycatch form, the position (latitude and longitude) and time of capture are recorded, if known. It is then possible to define at which point in the trawl or set the 'incident' occurred. It is also possible to have separate incidents for the same station, distinguished by *time_c* (time caught attribute). Observers can often ascertain the time of capture of a non-fish species, for example at the end of a tow, in which case the capture positions will be the same as the end of the tow. In cases where the position and time caught are not known, the position and time caught fields in the *t_nonfish_station* table are null. The start, end positions, and times of tows or sets are held in the *t_station* table.

The *t_nonfish_station* table also stores data specific to non-fish bycatch and environmental data requested on the non-fish bycatch form, along with some additional data from the log-book data (**obs** database), which is requested for non-fish captures and not stored elsewhere in **obs_lfs**.

The *t_nonfish_catch* table stores a record for each specimen caught. The species and sex recorded by the observer are stored in fields *species_obs* and *sex_obs*. As observers do not record the sex of birds, *sex_obs* will be null for marine birds. The species and sex fields are used to enter positive identifications (as obtained from subsequent autopsy reports where available); these two fields are otherwise left null.

3.6 Squid

The data collected by scientific observers on New Zealand arrow squid from both squid trawlers and jiggers, up to the end of the 2000/2001 fishing year, were held in a separate **squid** database.

Both the **obs_lfs** and **squid** databases stored sub-sets of station data collected by scientific observers, hence the squid data have been incorporated into the **obs_lfs** database.

Station data from the *t_station_squid* table in the **squid** database have been inserted into the GENSPEC structure for station data in **obs_lfs**. Attributes common to all station types are now stored in the *t_station* table. The specialised attributes from the *t_station_squid* table are now all stored in the *t_trawl* table; this includes data from both squid trawlers and squid jiggers. The squid jiggers can be distinguished in the same fashion as previously, using fishing method code. The information that was stored in the gear_meth attribute, in the *t_station_squid* table, is now held in the fishing_method attribute in *t_station*.

Weight data stored in the *t_station_squid* table in the **squid** database, have been inserted to the *t_general* table in the **obs_lfs** database. The total estimated green weight of squid stored in the attribute *species_weight*, is now stored in *catch_weight* attribute of *t_general*. The weight of measured squid, from the *wt_meas* attribute, is now stored in *sample_weight* attribute of *t_general*. The total number of squid measured, stored in *no_meas* has not been transferred to **obs_lfs**, as this number can be derived from the individual squid measurements. The species code in the *t_general* table was set to "SQU", for all squid samples because the total estimated green weight in the *t_station_squid* table, was summed from SQU, NOS & NOG codes. (Species code was not an attribute in the *t_station_squid* table). The *sample_weight* as recorded on the squid length frequency form, should refer to one species, because a new page should be completed for each species sampled by the observer. This was not maintained separately in the **squid** database. In practice, there are only two trips where both NOS and NOG (*Nototodarus sloanii,* and *N. gouldi)* have been sampled from the same station. Trip 512; there are 39 samples where the sample weight was combined, and trip 51 there are 2 stations, where it is unknown how the weight was recorded, but there was only one specimen of NOG in each of the two tows. For all other samples, the sample weight therefore refers to the only species sampled for the station.

The biological data for individual squid specimens are held in the table *t_fish_bio_squ*.
Previously, this was the table *t_fish_bio_asq* in the **squid** database.

3.7 Squid Jigger Technical Specification

The **squid** database also contained technical data on most licensed and some foreign chartered and domestic squid jiggers fishing in New Zealand waters. This information mainly covers the period from 1978 to 1988, with very little new information since 1988. The latest data are for the 1990/1991 fishing year. These data are held in the table *t_jig_specs*.

3.8 Seabird Warp_Strike Observations

A sampling programme to collect "Seabird Warp_Strike Observations in New Zealand trawl fisheries" was implemented by the observer programme in January 2005 for the Squid fishery. Tables to store the data collected have been created in the **obs_lfs** database (March 2006) and the data from 2005 onwards is loaded into these tables.

There are five related tables used to store the Seabird Warp-Strike Observations data; *t_warp_strike* holds the descriptors of the trawl being observed,
t_warp_strike_sample stores seabird warp/mitigation device strike observations and bird abundance data for each "fifteen-minute" sample period**.** The table *t_bird_capture* stores the total numbers of seabirds recovered from warps, net, mitigation devices or unknown sources for the whole tow. The table *t_warp_strike_devices* stores details of any mitigation devices or methods used during an observation sampling period and the table *t_mitigation_descriptions*

holds a detailed description for each distinct 'brief' description of mitigation devices or methods stored in the *t_warp_strike_devices* table. Several fields that are recorded at the trawl level on the form are stored at the sampling level in the database; 'observer initials' as cases of two observers undertaking independent observations (recorded on separate forms) for an individual trawl have occurred and the 'side observed' field, although instruction are for the same side to be observed for the whole trip, observations have been carried out on both the port and starboard sides during a single tow.

There have been various versions of the form "Seabird Warp-Strike Observations (Trawl)", with changes to information collected, therefore some attributes are not always be populated in the database. The large and small bird abundance counts were initially recorded as one of four ranges on the first version of the form. While large_range and small_range fields for the later forms are populated from the counts in large_birds and small_birds on later versions of the forms, actual abundance numbers for the earlier forms obviously cannot be derived and remain null. Recording of sprags on each warp and grease on warps are not recorded on the 18/01/2006 version of the form. The pre-recorded devices on the forms have changed, only 'tori line' and 'bird baffler' are recorded across all versions. The 18/01/2006 version of the form added a "To specification?" question for 'tori line', 'warp scarer' and 'bird baffler', this information has been incorporated into the deployed code for each device. The tori line details of length, height and "number of streamers" is no longer recorded on the latest form.

Note there are fields in this dataset where observers have not recorded data or not answered questions, that could be interpreted as zero or a continuation of previous entries for the same field, these fields have been retained as nulls, as it would not be possible to later distinguish such changes and therefore it is left to individual users to make their own interpretations. Errors that can be changed with certainty, such as dates or times are corrected. Note the 2005 data was loaded from electronic data supplied to NIWA from the Ministry of Fisheries.

3.9 Purse Seine Observations

The Observer Programme extended coverage of observed fishing activities to include the collection of data from purse-seine fishing trips, this data series commences in December 2004. The observers fill in two form types to capture effort, a "Vessel Activity Log" and a separate "Observer Catch Effort Set Details" form, data from these forms are stored in the **obs_lfs** database. There is some overlap of data collected on each form, e.g. target species, FMA, spotter plane call-sign, position. The combined information collected on these purse seine forms is stored using the GENSPEC structure (GENeralisation / SPEcialisation) as set up for storing trawl and BLL data. Fields including date, start and finish time, latitude, longitude, method, target species, FMA and seabed depth that are general to other fishing types are stored in the *t_station* table, while the data fields specific to purse-seine only (specialised entities), such as the activity code, spotter plane call-sign and times relating to the set operation are stored in the *t_purseseine* table. Each line of information that is recorded on the Vessel Activity Log is stored as a record in the *t_station* table, hence is assigned a sequential station number, a separate set number as recorded by observers when a set is made, is also stored as set_no in the *t_purseseine* table. Several views, *v_ps_set* and *v_activity* as detailed in sections 6.16 and 6.17 are designed to allow users to view details as recorded on the respective two purse-seine forms. Catch details are stored in the *t_catch* table and sampling data in the standard *t_general* and *t_lth* tables.

4 Table Summaries

The **obs_lfs** database is broken down into a set of tables and views containing length frequency information for the main ITQ species and bycatch species. The relevant trip and tow

information is downloaded from the **obs** database. The following is a listing and outline of the major tables and views contained within **obs_lfs**.

1**. t_trip :** provides a view of the details of observer trips for which length frequency data were collected, including vessel key, observer names, and start date.

a) **t_trip_master**: underlying table of the *t_trip* view restricted to the DBA only.

2. **t_trip_comm.** : general comments made on an observed trip, either by the observer or the database administrator.

3. **t_station** : stores details common to both trawl (sampled) and longline and purse seine sets, including date, depth, and position of the tow.

a) **v_station** View of t_station with position fields decimal degrees (to 0.1 deg)

4**. t_trawl :** provides details of the tows for each trip for which length frequency data were collected, that only relate to trawl and are not held in *t_station*.

a) **v_tow** : is a view that links data from *t_station* and *t_trawl*. It has the equivalent fields to those that were previously contained in the old table *t_tow*.

b) **v_tow_sop** : is a view that links data from *t_station* and *t_trawl*, for SOP data.

c) **v_tow_orm** : is a view that links data from *t_station* and *t_trawl,* for ORM data.

d) **v_tow_hmc** : is a view that links data from *t_station* and *t_trawl*. For HMC data.

5. **t_line** : provides details of the sets, that only relate to longline and are not held in *t_station*.

a) **v_line** : is a view that links data from *t_station* and *t_line*.

6. **t_catch** : Contains catch data per set, for all lining and purse seine sets.

7. **t_general :** contains the catch and sample weights of a given species from which length frequencies were obtained. A series of views for the major species emanate from this table:

a) **v_hoki :** contains data from *t_general* for hoki only.

b) **v_jma :** contains data from *t_general* for the three main jack mackerel species (*Trachurus declivis*, *T. murphyii,* and *T. novazelandae*) only.

c) **v_middepth :** contains data from *t_general* for the middle depth species only, including hake, ling, silver warehou, and gemfish only.

d) **v_oreo :** contains data from *t_general* for black and smooth oreos only.

e) **v_orh :** contains data from *t_general* for orange roughy only.

f) **v_sbw :** contains data from *t_general* for southern blue whiting only

g)**v_station_squid** : view of squid stations from t_station

h)**v_nog** : view data from t_fish_bio_squ for squid NOS *Nototodarus gouldi* only.

i) **v_nos** : view data from t_fish_bio_squ for squid NOS *N. sloanii* only.

8. **t_lth :** contains the length frequency data and the gonad staging data (using 5 gonad maturity stages, female fish only) from length frequency forms.

9. **t_nonfish_station** : contains details for stations with non-fish bycatch including extra parameters taken from the vessel's tow log

10. **t_nonfish_catch** : catch and biological details of non-fish bycatch

11. **t_fish_bio_squ**: contains biological data for individual squid specimens.

12. **t_jig_specs:** contains data relating to technical specifications of squid jiggers.

13**. t_warp_strike** contains data from seabird warp-strike observations (trawl) for the trawl being observed.

14**. t_warp_strike_sample** Fifteen minute seabird warp/mitigation device strike observations and bird abundance data**.**

15. **t_bird_capture** Contains the numbers of seabirds recovered from the whole tow.
16. **t_warp_strike_devices** Contains details of any mitigation devices or methods used during an observation sampling period
17. **t_mitigation_descriptions** Contains ddescriptions of mitigation devices or methods used.

5 obs_lfs Tables

The following are the main tables within the **obs_lfs** including attribute names, data types (and any range restrictions), and comments. Vessel keys assigned by the Ministry of Fisheries, have replaced the *vessel* (name) and *callsign* attributes in *t_trip*; this was implemented in August 2001. The table *t_trip* is now actually a view of the table *t_trip_master*, as shown in the Entity Relationship Diagram (ERD) in Figure 1.

5.1 Table 1: t_trip

Comment: Details of the trip including start and finish dates.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observed trip.
vessel_key	integer		The MFish id key assigned to this vessel.
nation	char(6,1)		Nation of origin of the vessel. Can also be nation codes for charter companies.
obs1	char(20,1)		Name of the first observer.
obs2	char(20,1)		Name of the second observer.
date_s	date(5)		Start date of the trip.
date_f	date(5)		Finish date of the trip.
origin	char(3,1)		Origin of the data.

Creator: sma

5.2 Table 2: t_trip_comm.

Comment: General comments made on an observed trip, either by the observer or the database administrator.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number.
comments	text(60,120,60,1)	No	General comments.

Creator: sma
Referential: Invalid Trip (trip_no) INSERT t_trip_master (trip_no)
Indices: UNIQUE BTREE trip_ndx ON (trip_no)

5.3 Table 3: t_station

Comment: Station data collected by observers, where a station is either a trawl or a longline set.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observer trip.
station_no	integer	No	A sequential number for each station of an observer trip.
date_s	date(5)		Start date of the tow or set.
target_sp	character(3,1)		3 character code for the species being targeted. Refer rdb:curr_spp.
time_s	integer		Start time (24 hour format, NZST).
lat_s	decimal(5,1)		Start position latitude (DDMM.m).
long_s	decimal(6,1)		Start position longitude (DDDMM.m).
ew_s	character(1,1)		Start position meridian, E or W.
area	character(4,1)		3 or 4 character area codes. Usually Fisheries Management Area codes, but also research codes where appropriate.
bottom_depth_s	integer		Depth (m) to the bottom (from either the net or the vessel) at the start of the tow.
date_f	date(5)		Finish date of the tow or 'Start of Haul' date for BLL.
time_f	integer		Finish time (24 hour format, NZST)
lat_f	decimal(5,1)		Finish position latitude (DDMM.m).
long_f	decimal(6,1)		Finish position longitude (DDDMM.m).
ew_f	character(1,1)		Finish position meridian, E or W.
bottom_depth_f	integer		Depth (m) to the bottom (from either the net or the vessel) at end of tow.
fishing_method	character(3,1)		3 character fishing method code

Creator: sma

Referential:	Invalid Trip (trip_no) INSERT t_trip_master (trip_no)
	Invalid area (area) INSERT rdb:area_codes (code)
Indices:	NORMAL (2, 15) ON (station_no)
	NORMAL (2, 15) ON (date_s)
	NORMAL (2, 15) ON (target_sp)
	NORMAL (2, 15) ON (bottom_depth_s)
	NORMAL (2, 15) ON (lat_s)
	UNIQUE BTREE ON (trip_no, station_no)

5.4 Table 4: t_trawl

Comment: Trawl details of a tow from which fish were sampled.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observer trip.
station_no	integer	No	A sequential number for each tow made during an observer trip.
gear	character(5,1)		Up to 5 character code for the type of fishing gear used for the tow.
headline	decimal(4,1)		Headline height (m) of the fishing gear during the tow.
net_depth_s	integer		Depth (m) of the trawl net at the start of the tow.
speed	decimal(3,1)		Mean speed (knots) during the tow.
net_depth_f	integer		Depth (m) of the trawl net at the end of the tow.
temp_surface	decimal(3,1)		Sea surface temperature (degrees Celsius).
temp_headline	decimal(3,1)		Sea temperature at headline (degrees Celsius).

Creator: sma
Referential: Invalid trip/tow (trip_no, station_no) INSERT t_station (trip_no, station_no)
Indices: NORMAL (2, 15) ON (station_no)
NORMAL (2, 15) ON (net_depth_s)
NORMAL (2, 15) ON (gear)
UNIQUE BTREE ON (trip_no, station_no)

5.5 Table 5: t_line

Comment: Details from a longline set and the corresponding haul of the set.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observer trip.
station_no	integer	No	A sequential number for each set made during an observer trip.
topography	smallint		A numeric code to describe the bottom contour.
no_hooks	integer		Number of hooks set.
bait1	character(3,1)		3 char species code for the principle bait species used.
bait2	character(3,1)		Species code for the 2nd most relevant bait species used.
percent_baited	smallint		Percentage of hooks that were baited.
lf	character(1,1)		Length Frequency done on fish from this set? Y = Yes, N = No.
hooks_lost	integer		Number of hooks lost
assessment	character(2,1)		Catch assessment for the degree of observation by the observer.
comments	text(120,120,120,1)		

Creator: sma
Referential: Invalid trip/set (trip_no, station_no) INSERT t_station (trip_no, station_no)
Indices: NORMAL (2, 15) ON (station_no)
UNIQUE BTREE ON (trip_no, station_no)

5.6 Table 6: t_catch

Comment: Contains catch data per set, for all longline sets.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observer trip.
set_no	integer	No	A sequential number for each set made during an observer trip.
species	character(3,1)		3 character code for a species caught from the set.
discard	character(1,1)		Discard status code.
catch_weight	longinteger		Weight (kg) of the catch of the species from the set.
no_fish	integer		Number of fish of this species in the catch.
w_meth	character(3,1)		Up to 3 character code for the method of obtaining catch weights at sea.
hold_no	character(3,1)		Hold number catch stored in.

Creator: sma
Referential: Invalid Trip (trip_no) INSERT t_trip (trip_no)
Invalid Species (species) INSERT rdb:curr_spp (code)
Indices: NORMAL (2, 15) BTREE ON (species)
NORMAL (2, 15) BTREE ON (set_no)
UNIQUE BTREE spp ON (trip_no, set_no, species, w_meth)

5.7 Table 7: t_general

Comment: Contains catch data by tow for all species used for sampling.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observer trip.
tow_no	integer	No	A sequential number for each tow made during an observer trip.
species	char(3,1)		3 character code for a species sampled from the tow.
sample_weight	decimal(4,1)		Weight (kg) of the sample taken from the whole catch of the tow.
sample_weigh_meth	integer		Integer code for the method of obtaining the sample weight: 1=Salter scales; 2=SeaWay scales; 3=Platform scales; 99=Miscellaneous code where other weighing method used or weight estimated.
catch_weight	longinteger		Weight (kg) of the catch of the species from the tow.
w_meth	char(3,1)		Up to 3 char code for the method of obtaining catch weights at sea.
lwm	integer		Unique integer code for the male length/weight regression parameters.
lwf	integer		Unique integer code for the female length/weight regression parameters.
lwt	integer		Unique integer code for the species length/weight regression parameters.

Creator: sma
Referential: Invalid Trip/station (trip_no,tow_no) INSERT t_station (trip_no,tow_no)
Invalid species (species) INSERT rdb:curr_spp (code)
Indices: UNIQUE BTREE general_pk ON (trip_no, tow_no, species)
NORMAL (2, 15) ON (tow_no)
NORMAL (2, 15) ON (species)

5.8 Table 8: t_lth

Comment: Contains length frequency data for a length class for any one species.

Attributes	Data Type	Null?	Comment
trip_no	longinteger	No	Trip identification number. A sequential number for each observer trip.
tow_no	integer	No	A sequential number for each tow made during an observer trip.
species	char(3,1)	No	3 character code for a species sampled from the tow.
meas_meth	char(1,1)		1 character code for the method of measuring length.
lgth	integer	No	Length class (lowest whole cm).
no_m	integer		Frequency of males in the length class.
no_f	integer		Frequency of females in the length class.
no_f1	shortinteger		Frequency of the female stage one gonads.
no_f2	shortinteger		Frequency of the female stage two gonads.
no_f3	shortinteger		Frequency of the female stage three gonads.
no_f4	shortinteger		Frequency of the female stage four gonads.
no_f5	shortinteger		Frequency of the female stage five gonads.
no_m1	shortinteger		Frequency of male stage one gonads.
no_m2	shortinteger		Frequency of male stage two gonads.
no_m3	shortinteger		Frequency of male stage three gonads.

no_m4	shortinteger		Frequency of male stage four gonads.
no_m5	shortinteger		Frequency of male stage five gonads.
no_t	integer	No	Frequency of all fish in the length class, including unsexed fish.

Creator: sma

Referential: Invalid Trip (trip_no) INSERT t_trip (trip_no)
Invalid species (species) INSERT rdb:curr_spp (code)
Invalid fish meas code (meas_meth) INSERT rdb:t_fish_meas_codes (fish_meas_code)
Invalid Station (trip_no, tow_no) INSERT t_station (trip_no, station_no)

Indices: NORMAL (2, 15) ON (tow_no)
NORMAL (2, 15) ON (species)
UNIQUE BTREE ON (trip_no, tow_no, species, lgth)

5.9 Table 9: t_nonfish_station

Comment: Details for stations with non-fish bycatch including extra parameters taken from the vessel's tow log

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observer trip.
tow_no	integer	No	A sequential number for each tow made during an observer trip.
time_c	integer		Time caught if known 24 hour format, NZST.
lat_c	decimal(5,1)		Exact position of capture if known.
long_c	decimal(6,1)		Exact position of capture if known.
ew_c	character(1,1)		Catch position meridian, E or W.
wingspread	integer		Distance between trawl wings.
depth_gear	integer		Depth of gear (m).
wind_knots	integer		Wind speed (knots).
wind_dir	integer		Wind direction (degrees) 0 to 359.
sea_state	integer		Sea state code - Beaufort scale.
cloud	integer		Cloud cover in eighths 0 = clear & 8 full cover.
offal	character(1,1)		0 = no, 1 = shooting only, 2 = on hauling only, 3 = on both, 9 = yes, undefined (pre trips 780).
tori_pole	character(1,1)		0 = no 1 = yes if a tori pole to CCAMLR specification used.
bird_device_o	character(1,1)		Was a bird scaring device used: 0 = no, 1 = yes.
gear_event	character(1,1)		0 = no, 1 = yes an event that affected the chance of catching a non-fish species took place.
bird_device_c	text(80,80,16,1)		Comments to specify a bird scaring device.

temp_surface	decimal(3,1)	Sea surface temperature (degrees Celsius).
temp_headline Celsius).	decimal(3,1)	Sea temperature at headline (degrees
tow_type	character(1,1)	1= bottom throughout, 2= midwater at relatively constant depth, 3= midwater in a broad range of depths, 4= mixed bottom & midwater.
tow_config	character(1,1)	Configuration of path of tow A=straight line, B="U", C=zigzag, D=closed pattern (circle, loop etc), E=constant depth contour, F=pinnacle fishing.
tow_turns	smallint	Number of turns during tow.
comments	text(120,120,16,1)	

Creator: sma
Referential: (trip_no) INSERT t_trip (trip_no)
Indices: NORMAL (2, 15) BTREE ON (trip_no)
NORMAL (2, 15) BTREE ON (tow_no)

5.10 Table 10: t_nonfish_catch:

Comment: Catch and biological details of non-fish bycatch

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observer trip.
tow_no	integer	No	A sequential number for each tow made during an observer trip.
time_c	integer		Time caught is repeated here to distinguish by-catch incidents.
specimen_no	integer		Number of species in tow (per incident defined by time caught).
species	character(3,1)		Species code. Remains null until positively identified in a post-mortem. Refer b:curr_spp
species_obs	character(3,1)	No	Observer species identification.
lgth	integer		Standard length for seals, Fork length for dolphins.
girth	integer		Girth (mm) at posterior margin of foreflippers.
blubber_mm	smallint		Blubber thickness (mm).
Sex	smallint		0 = unsexed, 1 = male, 2 = female Remains null until positively identified in a post-mortem.
sex_obs	smallint		Observer sex identification.
age	character(7,1)		Maturity for seabirds A=adult, AB=adult breeder, AN=adult nonbreeder, SA=subadult, I=immature, J=juvenile. Age mammals, estimated e.g. growth increments in years.
age_actual	character(7,1)		Actual age for marine mammals.
tag_no	character(15,1)		Tag or band number.
alive	integer		1= alive, 2= dead, 3= killed, 4= decomposing.

marked	character(1,1)	R= retained, D= discarded unmarked, M=Marked or tagged & discarded.
whole	character(1,1)	0= no, 1= yes whole specimen kept.
head	character(1,1)	0= no, 1= yes head kept.
leg	character(1,1)	0= no, 1= yes leg kept.
ovary	character(1,1)	0= no, 1= yes ovary sample taken.
stomach	character(1,1)	0= no, 1= yes stomach sample taken.
teeth	character(1,1)	0= no, 1= yes teeth collected.
skin	character(1,1)	0= no, 1= yes skin sample taken.
blubber	character(1,1)	0= no, 1= yes blubber sample taken.
muscle	character(1,1)	0= no, 1= yes muscle sample taken.
other	character(1,1)	0= no, 1= yes other sample taken (specified in comments).
observed	character(1,1)	0 No, 1 = yes observed caught species during fishing around vessel.
no_seen	integer	Number of species seen if observed during tow/set, recorded once against first specimen recorded.
remarks	text(40,40,16,1)	

Creator: sma
Referential: (trip_no, tow_no) INSERT t_nonfish_station (trip_no, tow_no)
Invalid species (species) INSERT rdb:curr_spp (code)
Invalid species obs (species_obs) INSERT rdb:curr_spp (code)
Indices: NORMAL (2, 15) ON BTREE (trip_no, tow_no)
NORMAL (2, 15) ON BTREE (species, alive, marked)

5.11 Table 11: t_fish_bio_squ:

Comment: This table contains biological data for individual squid specimens sampled by Scientific Observers. Data are linked to table *t_station* by the combination of trip number and tow number and to tables in **obs** by the same.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Observer trip number.
tow_no	integer	No	Sequential identifier for each tow.
species	character(3,1)		Species codes
fish_no	integer	No	Identifies the individual specimen.
sex	smallint		0=unsexed, 1=male, 2=female.
Copul	smallint		Female only;0=not copulated and 1=copulated.
lgth	smallint		Dorsal mantle length (DML) in cm.

Creator: sma
Referential: No station data for tow (trip_no, tow_no) INSERT t_station_squid (trip_no, tow_no)
Indices: NORMAL (2, 15) BTREE i_tripnumber ON (trip_no)
NORMAL (2, 15) BTREE i_townumber ON (tow_no)
NORMAL (2, 15) BTREE i_species ON (species)
NORMAL (2, 15) BTREE ON (sex)
UNIQUE BTREE i_squid ON (trip_no, tow_no, species, fish_no)

5.12 Table 12: t_jig_specs:

Comment: This table contains data relating to technical specifications of squid jiggers. Data were recorded from fishing licence applications - complete data n/a after 8788 (foreign chartered and domestic only)

Attributes	Data Type	Null?	Comment
fishing_yr	integer		Fishing year, eg. 8788 (= Oct 87 to Sep 88).
Nation	character(6,1)		Nationality of vessel,eg. JAPAN (= Japan licensed).
call_sign	character(6,1)	No	Vessel callsign.
lgth	decimal(5,2)		Registered length of vessel (metres to 2 decimals).
wt	decimal(5,2)		Gross weight (tonnes to 2 decimals).
speed_s	decimal(3,1)		Service or normal speed (knots to 1 decimal).
speed_m	decimal(3,1)		Maximum speed (knots to 1 decimal).
duration	integer		Maximum duration at sea (days).
gen1	smallint		Number of generators of attribute "kva1" power.
kva1	integer		Power of attribute "gen1" generators (kva).
gen2	smallint		Number of generators of attribute "kva2" power.
kva2	integer		Power of attribute "gen2" generators (kva).
gen3	smallint		Number of generators of attribute "kva3" power.
kva3	integer		Power of attribute "gen3" generators (kva).
gen4	smallint		Number of generators of attribute "kva4" power.
kva4	integer		Power of attribute "gen4" generators (kva).
hold_cap	integer		Total fish hold capacity (tonnes).

freezer	smallint	Capacity of quick or blast freezers (tonnes/day).
jigs_h	smallint	Number of hand jig machines.
jigs_m	smallint	Number of automatic jig machines.
lures	smallint	Number of lures per line.
Dist_l	decimal(2,1)	Distance between lures (metres).
light1	integer	Number of lights of attribute "w1" wattage.
watts1	integer	Power of attribute "light1" lights (watts).
light2	integer	Number of lights of attribute "w2" wattage.
watts2	integer	Power of attribute "light2" lights (watts).
light3	integer	Number of lights of attribute "w3" wattage.
watts3	integer	Power of attribute "light3" lights (watts).
light4	integer	Number of lights of attribute "w4" wattage.
watts4	integer	Power of attribute "light4" lights (watts).

Creator: sma
Indices: NORMAL (2, 15) BTREE i_call_sign ON (call_sign)
NORMAL (2, 15) BTREE i_date ON (fishing_yr)
UNIQUE BTREE i_both ON (fishing_yr, call_sign)

5.13 Table 13: t_warp_strike

Comment: Seabird warp-strike observations (trawl); fishing event descriptors for the trawl being observed.

Attributes	Data Type	Null?	Comment
stn_key	longinteger	No	Primary key for unique identification of the trawl warp-strike observations.
trip_no	integer	No	Trip identification number for an observer trip.
station_no	integer	No	A sequential number for each station of an observer trip.
tcepr_no	longinteger		TCEPR form number.
tcepr_tow	smallint		Shot number on the TCEPR form.
tow_date	date(0)		Start date of the trawl
tow_start	integer		Start time of the tow defined as when the doors go into the water (hhmm).
time_code	character(2,1)		Time code as defined in Observer Catch Effort Logbook Instructions.
tow_end	integer		End time of the tow defined as when the doors leave the water (hhmm).
meal_plant	character(1,1)		Meal plant onboard the vessel (Y or N).
meal_plant_on	character(1,1)		Meal plant running during the tow (Y or N).
percent_obs	smallint		The percent of pound emptying observed.
comments_tow	character(160,1)		Comment for the tow or relating to a sampling period that was not sampled, therefore does not have an entry in the *t_warp_strike_sample* table.

Creator: dba
Indices: UNIQUE TIMESERIES indx stn key ON (stn_key)
UNIQUE TIMESERIES indx trip stn ON (trip_no, station_no)

5.14 Table 14: t_warp_strike_sample

Comment: Fifteen minute seabird warp/mitigation device strike observations and bird abundance data.

Attributes	Data Type	Null?	Comment
sample_key	longinteger	No	Primary key for unique identification of t_warp_strike_sample records.
stn_key	longinteger	No	Foreign key to link to trawl details in t_warp_strike.
trip_no	integer	No	Trip identification number for an observer trip.
station_no	integer	No	A sequential number for each station of an observer trip.
sample_no	smallint	No	Sampling period number for the tow
side_observed	character(1,1)		The warp or side observed during the observation sampling period P=Port, S=Starboard, C=Central.
observed	character(2,1)		Code for trawl warp (TW) or mitigation device (MD) observed during the sampling period. TM = trawl & mitigation
large_birds	integer		The large bird abundance count for the sampling period.
small_birds	integer		The small bird abundance count for the sampling period.
large_range	smallint		Code for the large bird abundance range for the sampling period; 0 = 0, 1 = 1 - 9, 2 = 10 -100, 3 = >100
small_range	smallint		Code for the small bird abundance range for the sampling period; 0 = 0, 1 = 1 - 9, 2 = 10 -100, 3 = >100.
time_start	integer		Start time for the sampling period (hhmm)
time_end	integer		End time for the sampling period (hhmm)

contacts_large	smallint	Number of heavy contacts for large birds with trawl warp or mitigation device for the sampling period.
contacts_small	smallint	Number of heavy contacts for small birds with trawl warp or mitigation device for the sampling period.
tori_line	character(1,1)	tori line used; Y = Yes (to specification), N = No, X = yes, not to specification
warp_scarer	character(1,1)	Warp scarer used; Y = Yes (to specification), N = No, X =Yes, not to specification
bird_baffler	character(1,1)	Bird baffler used; Y = Yes (to specification), N = No, X = Yes, not to specification
sonic_scarer	integer	Number of times a high frequency sonic device was activated during the sampling period, 0 = not used or not present.
gas_canon	integer	Number of times a gas canon was activated during the sampling period, 0 = not used or not present.
other_desc	character(20,1)	Other mitigation description.
sprags_port	character(1,1)	Sprags on portside warp; Y = Yes, N = No, U = Unknown.
sprags_starboard	character(1,1)	Sprags on starboard warp; Y = Yes, N = No, U = Unknown.
grease	character(1,1)	Grease on warps; P = Port, S = Starboard, B = Both, N = Neither / None.
swell_ht	decimal(3,2)	Swell height (m)
swell_dir	smallint	Swell direction, 12 point “clock” scale relative to vessel direction (1-12 h)
wind_spd	smallint	Wind speed (Beaufort)
wind_dir	smallint	Wind direction, 12 point “clock” scale relative to vessel direction (1-12 h)
discharge_side	character(1,1)	Discharge side code; P=Port, S=Starboard, B=Both, N=Neither/None

discharge_rate	character(1,1)	Discharge rate; 0=none, 1 = negligible, 2 = intermittent, 3 = continuous
discharge_type	character(5,1)	Discharge types, each applicable code included in this field; S=Sump water, M = Minced, C= cutter pump, O= Offal, D=Discards
obs_initials	character(2,1)	Observer initials
comments	character(280,1)	Comments for the sampling period.

Creator: dba
Referential: Invalid trip station numbers (trip_no, station_no)
INSERT t_warp_strike (trip_no, station_no)
Invalid stn key (stn_key) INSERT t_warp_strike (stn_key)
Indices: UNIQUE TIMESERIES indx sample key ON (sample_key)
NORMAL (2, 15) BTREE indx trip ON (trip_no)
NORMAL (2, 15) BTREE indx station ON (station_no)
NORMAL (2, 15) BTREE indx stn key ON (stn_key)

5.15 Table 15: t_bird_capture:

Comment: Numbers of seabirds recovered from trawl warps, net, mitigation devices or unknown sources for the whole tow.

Attributes	Data Type	Null?	Comment
bcap_key	longinteger	No	Primary key to identify bird capture records
stn_key	longinteger	No	Foreign key links t_bird_capture records to trawl details in t_warp_strike
recov_from	character(1,1)		Code for where the birds were recovered from; W=Warp, N=Net, M=Mitigation device, U=Unknown sources
status	character(1,1)		Code for the status; D=dead, I=injured, A=non injured, U=Unknown when no observation was made; size and bird count should be null when status is U.
size	character(1,1)		Code for bird size; L=Large, S=Small, N=Not recorded (pre 18/01/2006 forms)
bird_count	smallint		Number of birds recovered

Creator: dba
Indices: UNIQUE TIMESERIES indx bcap key ON (bcap_key)
NORMAL (2, 15) BTREE indx stn key ON (stn_key)

5.16 Table 16: t_warp_strike_devices

Comment: Details of any mitigation devices or methods used during an observation sampling period

Attributes	Data Type	Null?	Comment
desc_key	integer	No	Primary key on t_warp_strike_devices
sample_key	longinteger	No	Foreign key links to t_warp_strike_sample
d_type	character(20,1)		Device type name
d_length	integer		Length parameter of the device
d_height	integer		Height parameter of the device
streamers	integer		Number of streamers (tori line)
d_complete	character(1,1)		Device complete; Y= Yes, N = No, U = Unknown.
deploy_sides	character(1,1)		Sides device deployed on; P=Port, S=Starboard, B=Both, N=Neither / None

Creator: dba
Referential: Invalid sample key (sample_key) INSERT t_warp_strike_sample (sample_key)
Indices: UNIQUE TIMESERIES indx desc key ON (desc_key)
NORMAL (2, 15) BTREE indx sample key ON (sample_key)

5.17 Table 17: t_mitigation_descriptions

Comment: Description of mitigation devices or methods used

Attributes	Data Type	Null?	Comment
mi_key	integer	No	Primary key on *t_mitigation_descriptions.*
d_type	character(20,1)		Brief description of the mitigation device, this is also the foreign key link to the t_warp_strike_devices table.
description	character(80,1)		Full description of the mitigation device

Creator: dba
Indices: UNIQUE TIMESERIES indx mi key ON (mi_key)

5.18 Table 18: t_purseseine

Comment: Details from Observer Programme Purse Seine Catch Effort form and vessel activity log.

Attributes	Data Type	Null?	Comment
trip_no	integer	No	Trip identification number. A sequential number for each observer trip.
station_no	integer	No	A sequential number for each station of an observer trip.
set_no	integer		A sequential number for each set of an observer purse seine trip.
trip_day	integer		Trip days since the day observer joined the vessel (first day = l)
activity	character(4,1)		Code for vessel activity
beaufort	smallint		Beaufort scale of wind force
sea_temp	decimal(3,1)		Sea temperature
school_assn	character(2,1)		Target school association with code eg 9 if saw birds feeding on the target school
detect	character(2,1)		Code for who intially detected the target school
aircraft	character(6,1)		Spotter aircraft call sign
begin_purse	integer		Time begin pursing (winch on) (hhmm)
end_purse	integer		Time end pursing (rings up) (hhmm)
net_rolling	integer		Time net rolling started (hhmm)
net_sacking	integer		Time net sacking began (hhmm)
begin_brail	integer		Time begin brailling (hhmm)
end_brail	integer		Time brailling ended (hhmm)
tot_gw_surface	longinteger		Total greenweight at surface kg
gw_meth_surface	character(3,1)		Total greenweight at surface assessment method

tot_gw_onboard	longinteger	Total greenweight onboard kg
gw_meth	character(3,1)	Total greenweight onboard assessment method
result	character(1,1)	Result of set code
brail_code	character(1,1)	Brail type code
tot_losses	longinteger	Total losses kg
loss_meth	character(3,1)	Loss weight method code
loss_code	character(1,1)	Loss code
loss_stage	character(2,1)	Loss event stage code
loss_time	integer	Time catch lost (hhmm)
mdbd	character(1,1)	Sampling MDBD (Y/N)
lf	character(1,1)	Sampling LF (Y/N)
birds_obs	character(1,1)	Sampling birds obs (Y/N)
nfb	character(1,1)	Sampling NFB (Y/N)
mammal	smallint	Non-fish bycatch mammal (number)
seabird	smallint	Non-fish bycatch seabird (number)
turtle	smallint	Non-fish bycatch turtle (number)
time_codes	character(9,1)	Time codes used for each time (nnnnnnn where n is either 1 or 2 for each time)
celr_no	longinteger	CELR No
port	character(12,1)	Port when activity requires a port or airport for aircraft
comments	character(200,1)	Comments from activity log
comment_ce	character(380,1)	Comments from Catch Effort form

Creator: dba
Indices: NORMAL (2, 15) BTREE set_indx ON (set_no)
NORMAL (2, 15) BTREE CELR_indx ON (celr_no)
UNIQUE BTREE trip_stn_indx ON (trip_no, station_no)

6 obs_lfs Views

The following are views that select data from *t_station* and either *t_trawl* or *t_set*. View *v_tow* gives a view of all the trawl related fields (equivalent to *t_tow* in the preceding report on the **obs_lfs** database). View *v_set* gives a view of all related set data.

6.1 View 1: v_tow

View: select attr 's'.'trip_no', attr 's'.'station_no', attr 's'.'date_s', attr 's'.'target_sp', attr t'.'gear', attr 't'.'headline', attr 's'.'time_s', convert deg2dec (attr 's'.'lat_s', 'S', 2) to decimal(3, 1) print 'lat_s' width 5, convert deg2dec (attr 's'.'long_s', attr 'ew_s', 2) to decimal(4, 1) print 'long_s' width 6, attr 's'.'area', attr 't'.'net_depth_s', attr 's'.'bottom_depth_s', attr 't'.'speed', attr 's'.'time_f', convert deg2dec (attr 's'.'lat_f', 'S', 2) to decimal(3, 1) print 'lat_f' width 5, convert deg2dec (attr 's'.'long_f', attr 'ew_f', 2) to decimal(4, 1) print 'long_f' width 6, attr 't'.'net_depth_f', attr 's'.'bottom_depth_f' from 't_station' 's', 't_trawl' 't' where ((attr 's'.'trip_no' = attr 't'.'trip_no' and attr 's'.'station_no' = attr 't'.'station_no'))

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
date_s	date(5)	
target_sp	character(3,1)	
gear	character(5,1)	
headline	decimal(4,1)	
time_s	integer	
lat_s	decimal(3,1)	
long_s	decimal(4,1)	
area	character(4,1)	
net_depth_s	integer	
bottom_depth_s	integer	
speed	decimal(3,1)	
time_f	integer	
lat_f	decimal(3,1)	
long_f	decimal(4,1)	
net_depth_f	integer	
bottom_depth_f	integer	

Creator: sma

6.2 View 2: v_line

View select attr 'trip_no', attr 'station_no', attr 'date_s', attr 'target_sp', attr 'time_s', convert deg2dec (attr 's'.'lat_s', 'S', 2) to decimal(3, 1) print 'lat_s' width 5, convert deg2dec (attr 's'.'long_s', attr 'ew_s', 2) to decimal(4, 1) print 'long_s' width 6, attr 'area', attr 'bottom_depth_s', attr 'date_f', attr 'time_f', convert deg2dec (attr 's'.'lat_f', 'S', 2) to decimal(3, 1) print 'lat_f' width 5, convert deg2dec (attr 's'.'long_f', attr 'ew_f', 2) to decimal(4, 1) print 'long_f' width 6, attr 'bottom_depth_f', attr 'fishing_method', attr 'topography', attr 'no_hooks', attr 'bait1', attr 'bait2', attr 'percent_baited', attr 'lf', attr 'hooks_lost', attr 'assessment', attr 'comments' from 't_station' 's', 't_line' 'l' where ((attr 's'.'trip_no' = attr 'l'.'trip_no' and attr 's'.'station_no' = attr 'l'.'station_no'))

Attributes	**Data Type**	**Null?**
trip_no	integer	No
set_no	integer	No
date_s	date(5)	
target_sp	character(3,1)	
time_s	integer	
lat_s	decimal(3,1)	
long_s	decimal(4,1)	
area	character(4,1)	
bottom_depth_s	integer	
date_f	date(5)	
time_f	integer	
lat_f	decimal(3,1)	
long_f	decimal(4,1)	
bottom_depth_f	integer	
fishing_method	character(3,1)	
topography	smallint	
no_hooks	integer	
bait1	character(3,1)	
bait2	character(3,1)	
percent_baited	smallint	
lf	character(1,1)	
hooks_lost	integer	
assessment	character(2,1)	
comments	text(120,120,120,1)	

Creator: sma

The following are views of the table *t_general*. Each view is a "filter" into this table, and reveals records for a selected species only.

6.3 View 3: v_hoki

View: select * from 't_general' where (attr 'species' = 'HOK')

Attributes	**Data Type**	**Null?**
trip_no	integer	No
tow_no	integer	No
species	char(3,1)	
sample_weight	decimal(4,1)	
sample_weigh_meth	integer	
catch_weight	longinteger	No
w_meth	char(3,1)	

Creator: sma

6.4 View 4: v_jma

View: select attr 'trip_no', attr 'tow_no', attr 'species', attr 'sample_weight', attr 'sample_weigh_meth', attr 'catch_weight', attr 'w_meth' from 't_general' where (attr 'species' match 'JM[AMND]')

Attributes	**Data Type**	**Null?**
trip_no	integer	No
tow_no	integer	No
species	character(3,1)	
sample_weight	decimal(5,1)	
sample_weigh_meth	integer	
catch_weight	longinteger	No
w_meth	character(3,1)	

Creator: sma

6.5 View 5: v_middepth

View: select attr 'trip_no', attr 'tow_no', attr 'species', attr 'sample_weight', attr 'sample_weigh_meth', attr 'catch_weight', attr 'w_meth' from 't_general' where (attr 'species' in ('HAK', 'LIN', 'SKI', 'SWA', 'SCI'))

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
species	char(3,1)	
sample_weight	decimal(5,1)	
sample_weigh_meth	integer	
catch_weight	longinteger	
w_meth	char(3,1)	

Creator: sma

6.6 View 6: v_oreo

View: select * from ‘t_general’ where (attr ‘species’ = ‘SSO’ or attr ‘species’ = ‘BOE’)

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
species	char(3,1)	
sample_weight	ecimal(4,1)	
sample_weigh_meth	integer	
catch_weight	longinteger	No
w_meth	char(3,1)	

Creator: sma

6.7 View 7: v_orh

View: select * from 't_general' where (attr 'species' = 'ORH')

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
species	char(3,1)	
sample_weight	decimal(4,1)	
sample_weigh_meth	integer	
catch_weight	longinteger	
w_meth	char(3,1)	

Creator: sma

6.8 View 8: v_sbw

View: select * from 't_general' where (attr 'species' = 'SBW')

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
species	char(3,1)	
sample_weight	decimal(5,1)	
sample_weigh_meth	integer	
catch_weight	longinteger	No
w_meth	char(3,1)	

Creator: sma

The following are views of the squid station data and squid biological measurements.

6.9 View 9 : v_station_squid

View: select attr 'trip_no', attr 'station_no', attr 'date_s', attr 'target_sp', attr 'time_s', attr 'lat_s', attr 'long_s', attr 'ew_s', attr 'area', attr 'bottom_depth_s', attr 'date_f', attr 'time_f', attr 'lat_f', attr 'long_f', attr 'ew_f', attr 'bottom_depth_f', attr 'fishing_method' from 't_station' where (attr 'fishing_method' in ('01', '02', '06', '08'))

Attributes	Data Type	Null?
trip_no	integer	No
station_no	integer	No
date_s	date(5)	
target_sp	character(3,1)	
time_s	integer	
lat_s	decimal(5,1)	
long_s	decimal(6,1)	
ew_s	character(1,1)	
area	character(4,1)	
bottom_depth_s	integer	
date_f	date(5)	
time_f	integer	
lat_f	decimal(5,1)	
long_f	decimal(6,1)	
ew_f	character(1,1)	
bottom_depth_f	integer	
fishing_method	character(3,1)	

Creator: sma

6.10 View 10: v_nog

View: select attr 'trip_no', attr 'tow_no', attr 'species', attr 'fish_no', attr 'sex', attr 'copul', attr 'lgth' from 't_fish_bio_asq' where (attr 'species' = 'NOG')

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
species	character(3,1)	
fish_no	integer	No
sex	smallint	
copul	smallint	
lgth	smallint	

Creator: sma

6.11 View: 11 v_nos

View: select attr 'trip_no', attr 'tow_no', attr 'species', attr 'fish_no', attr 'sex', attr 'copul', attr 'lgth' from 't_fish_bio_asq' where (attr 'species' = 'NOS')

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
species	character(3,1)	
fish_no	integer	No
sex	smallint	
copul	smallint	
lgth	smallint	

Creator: sma

The following are views that select data from *t_station* and *t_trawl*. Each view is a filter of all combined station data for trips with an origin of SOP, ORM, HMC respectively.

6.12 View: 11 v_tow_sop

View: select attr 's'.'trip_no', attr 's'.'station_no', attr 's'.'date_s', attr 's'.'target_sp', attr 't'.'gear', attr 't'.'headline', attr 's'.'time_s', convert deg2dec (attr 's'.'lat_s', 'S', 2) to decimal(3, 1) print 'lat_s' width 5, convert deg2dec (attr 's'.'long_s', attr 'ew_s', 2) to decimal(4, 1) print 'long_s' width 6, attr 's'.'area', attr 't'.'net_depth_s', attr s'.'bottom_depth_s', attr 't'.'speed', attr 's'.'time_f', convert deg2dec (attr 's'.'lat_f', 'S', 2) to decimal(3, 1) print 'lat_f' width 5, convert deg2dec (attr 's'.'long_f', attr 'ew_f', 2) to decimal(4, 1) print 'long_f' width 6, attr 't'.'net_depth_f', attr 's'.'bottom_depth_f'from 't_station' 's', 't_trawl' 't', 't_trip' 'r' where ((((attr 'r'.'origin' = 'SOP' and attr 'r'.'trip_no' = attr 's'.'trip_no') and attr 's'.'trip_no' = attr 't'.'trip_no') and attr 's'.'station_no' = attr 't'.'station_no'))

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
date_s	date(5)	
target_sp	character(3,1)	
gear	character(5,1)	
headline	decimal(4,1)	
time_s	integer	
lat_s	decimal(3,1)	
long_s	decimal(4,1)	
area	character(4,1)	
net_depth_s	integer	
bottom_depth_s	integer	
speed	decimal(3,1)	
time_f	integer	
lat_f	decimal(3,1)	
long_f	decimal(4,1)	
net_depth_f	integer	
bottom_depth_f	integer	

Creator: dba

6.13 View: 13 v_tow_orm

View: select attr 's'.'trip_no', attr 's'.'station_no', attr 's'.'date_s', attr 's'.'target_sp', attr 't'.'gear', attr 't'.'headline', attr 's'.'time_s', convert deg2dec (attr 's'.'lat_s', 'S', 2) to decimal(3, 1) print 'lat_s' width 5, convert deg2dec (attr 's'.'long_s', attr 'ew_s', 2) to decimal(4, 1) print 'long_s' width 6, attr 's'.'area', attr t'.'net_depth_s', attr 's'.'bottom_depth_s', attr 't'.'speed', attr 's'.'time_f', convert deg2dec (attr 's'.'lat_f', 'S', 2) to decimal(3, 1) print 'lat_f' width 5, convert deg2dec (attr 's'.'long_f', attr 'ew_f', 2) to decimal(4, 1) print 'long_f' width 6, attr t'.'net_depth_f', attr 's'.'bottom_depth_f' from 't_station' 's', 't_trawl' 't', 't_trip' 'r' where ((((attr 'r'.'origin' = 'ORM' and attr 'r'.'trip_no' = attr 's'.'trip_no') and attr 's'.'trip_no' = attr 't'.'trip_no')and attr 's'.'station_no' = attr 't'.'station_no'))

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
date_s	date(5)	
target_sp	character(3,1)	
gear	character(5,1)	
headline	decimal(4,1)	
time_s	integer	
lat_s	decimal(3,1)	
long_s	decimal(4,1)	
area	character(4,1)	
net_depth_s	integer	
bottom_depth_s	integer	
speed	decimal(3,1)	
time_f	integer	
lat_f	decimal(5,1)	
long_f	decimal(6,1)	
ew_f	character(1,1)	
net_depth_f	integer	
bottom_depth_f	integer	

Creator: dba

6.14 View 14: v_tow_hmc

View: select attr 's'.'trip_no', attr 's'.'station_no', attr 's'.'date_s', attr 's'.'target_sp', attr 't'.'gear', attr 't'.'headline', attr 's'.'time_s', convert deg2dec (attr 's'.'lat_s', 'S', 2) to decimal(3, 1) print 'lat_s' width 5, convert deg2dec (attr 's'.'long_s', attr 'ew_s', 2) to decimal(4, 1) print 'long_s' width 6, attr 's'.'area', attr 't'.'net_depth_s', attr 's'.'bottom_depth_s', attr 't'.'speed', attr 's'.'time_f', convert deg2dec (attr 's'.'lat_f', 'S', 2) to decimal(3, 1) print 'lat_f' width 5, convert deg2dec (attr 's'.'long_f', attr 'ew_f', 2) to decimal(4, 1) print 'long_f' width 6, attr 't'.'net_depth_f', attr 's'.'bottom_depth_f' from 't_station' 's', 't_trawl' 't', 't_trip' 'r' where ((((attr 'r'.'origin' = 'HMC' and attr 'r'.'trip_no' = attr 's'.'trip_no') and attr 's'.'trip_no' = attr t'.'trip_no') and attr 's'.'station_no' = attr 't'.'station_no'))

Attributes	Data Type	Null?
trip_no	integer	No
tow_no	integer	No
date_s	date(5)	
target_sp	character(3,1)	
gear	character(5,1)	
headline	decimal(4,1)	
time_s	integer	
lat_s	decimal(3,1)	
long_s	decimal(4,1)	
area	character(4,1)	
net_depth_s	integer	
bottom_depth_s	integer	
speed	decimal(3,1)	
time_f	integer	
lat_f	decimal(3,1)	
long_f	decimal(4,1)	
net_depth_f	integer	
bottom_depth_f	integer	

Creator: dba

6.15 View 15: v_station

View: select attr 'trip_no', attr 'station_no', attr 'date_s', attr 'target_sp', attr 'time_s', convert deg2dec (attr 'lat_s', 'S', 2) to decimal(3, 1) print 'lat_s' width 5, convert deg2dec (attr 'long_s', attr 'ew_s', 2) to decimal(4, 1) print 'long_s' width 6, attr 'area', attr 'bottom_depth_s', attr 'date_f', attr 'time_f', convert deg2dec (attr 'lat_f', 'S', 2) to decimal(3, 1) print 'lat_f' width 5, convert deg2dec (attr 'long_f', attr 'ew_f', 2) to decimal(4, 1) print 'long_f' width 6, attr ‘bottom_depth_f', attr ‘fishing_method' from 't_station'

Attributes:

trip_no	integer Not Null
station_no	integer Not Null
date_s	date(5)
target_sp	character(3,1)
time_s	integer
lat_s	decimal(3,1)
long_s	decimal(4,1)
area	character(4,1)
bottom_depth_s	integer
date_f	date(5)
time_f	integer
lat_f	decimal(3,1)
long_f	decimal(4,1)
bottom_depth_f	integer
fishing_method	character(3,1)

Creator: dba

6.16 View 16: v_ps_set

View: select attr 'a'.'trip_no', attr 'b'.'set_no', attr'a'.'date_s', attr 'a'.'time_s', attr 'a'.'time_f', attr 'a'.'lat_s', attr 'a'.'long_s', attr 'a'.'ew_s', attr 'b'.'sea_temp', attr 'a'.'bottom_depth_s', attr 'b'.'beaufort', attr 'b'.'school_assn', attr 'b'.'detect', attr 'a'.'target_sp', attr 'a'.'area' print 'fma', attr 'b'.'aircraft', attr 'b'.'begin_purse', attr 'b'.'end_purse', attr 'b'.'net_rolling', attr 'b'.'begin_brail', attr 'b'.'end_brail', attr 'b'.'tot_gw_surface', attr 'b'.'tot_gw_onboard', attr 'b'.'comment_ce' from 't_station' 'a', t_purseseine' 'b' where (((attr 'b'.'activity' in ('1', 'X1') or not attr 'b'.'set_no' = null) and (attr 'a'.'trip_no' = attr 'b'.'trip_no' and attr 'a'.'station_no' = attr 'b'.'station_no')))

Attributes:

trip_no	integer Not Null
set_no	integer
date_s	date(5)
time_s	integer
time_f	integer
lat_s	decimal(5,1)
long_s	decimal(6,1)
ew_s	character(1,1)
sea_temp	decimal(3,1)
bottom_depth_s	integer
beaufort	smallint
school_assn	character(2,1)
detect	character(2,1)
target_sp	character(3,1)
fma	character(4,1)
aircraft	character(6,1)
begin_purse	integer
end_purse	integer
net_rolling	integer
begin_brail	integer
end_brail	integer
tot_gw_surface	longinteger
tot_gw_onboard	longinteger
comment_ce	character(380,1)

Creator: dba

6.17 View 17: v_activity

View: select attr 'a'.'trip_no', attr 'a'.'station_no', attr 'b'.'set_no', attr 'b'.'trip_day', attr 'a'.'date_s', attr 'b'.'activity', attr 'a'.'time_s', attr 'a'.'time_f', attr 'a'.'lat_s', attr 'a'.'long_s', attr 'a'.'ew_s', attr 'b'.'port', attr 'b'.'beaufort', attr 'b'.'school_assn', attr 'b'.'detect', attr 'a'.'target_sp', attr 'a'.'area' print 'fma', attr 'b'.'aircraft', attr 'b'.'comments' from 't_station' 'a', 't_purseseine' 'b' where ((attr 'a'.'trip_no' = attr 'b'.'trip_no' and attr 'a'.'station_no' = attr 'b'.'station_no'))

Attributes:

trip_no	integer Not Null
station_no	integer Not Null
set_no	integer
trip_day	integer
date_s	date(5)
activity	character(4,1)
time_s	integer
time_f	integer
lat_s	decimal(5,1)
long_s	decimal(6,1)
ew_s	character(1,1)
port	character(12,1)
beaufort	smallint
school_assn	character(2,1)
detect	character(2,1)
target_sp	character(3,1)
fma	character(4,1)
aircraft	character(6,1)
comments	character(200,1)

Creator: dba

7. obs_lfs business rules

7.1 Introduction to business rules

The following are a list of business rules applying to the **obs_lfs** database. A business rule is a written statement specifying what the information system must do or how it must be structured. In this instance the information system is any system that is designed to handle observer length frequency sampling data.

There are three recognised types of business rules:

Fact	Certainty or an existence in the information system.
Formula	Calculation employed in the information system.
Validation	Constraint on a value in the information system.

Fact rules are shown on the ERD by the cardinality; e.g., one-to-many, of table relationships. Formula and Validation rules are implemented by referential constraints, range checks, and algorithms both in the database and during validation.

Validation rules may be part of the preloading checks on the data as opposed to constraints or checks imposed by the database. These rules sometimes state that a value should be within a certain range. All such rules containing the word 'should' are conducted by preloading software. The use of the word 'should' in relation to these validation checks means that a warning message is generated when a value falls outside this range and the data are then checked further in relation to this value. Hence in a small number of cases values may legitimately be outside the range of business rules containing the word 'should'.

7.2 Summary of rules

Observer trip record (t_trip)

trip_no Must be a unique integer.

vessel_key Must be a valid vessel key of the vessel observed, as entered into the **obs** database.

date_s The start date of the trip must be a legitimate date within the specified period the data set covers.

date_f The finish date of the trip must be a legitimate date within the specified period the data set covers.

Multiple column checks on date:
The start date must not be later than the finish date. The dates should be within a period of six weeks of each other.

origin Must be a valid origin code as listed in Appendix 1.

Observer trip comment record (t_trip_comm)

trip_no Must be equal to a trip number as listed in the *t_trip* table.

Observer station record (t_station)

trip_no Must be equal to a trip number held in the *t_trip* table.

station_no Must be a unique integer within all station records, for a given trip number.

date_s The start date of the station must be a legitimate date.

Multiple column checks on station start date, trip start date and trip finish date:
The station start date must fall within the range of the trip start and finish dates. The station start date should be sequential between stations, for a given trip.

target_sp Must be a valid species code as listed in the *curr_spp* table in the **rdb** database.

time_s Station start time must be a valid 24-hour time of between 0000 - 2359.

lat_s Must be a valid latitude and degrees should fall within the range of 33 - 56 South.

long_s Must be a valid longitude and degrees should fall within the range of 164 East to 170 West.

ew_s Longitude East or West at start, must be either "E" or "W".

area Must be one of the valid area codes for the New Zealand Exclusive Economic Zone (EEZ) or a valid "ET' area code (outside zone), as listed in Appendix 1.

bottom_depth_s Bottom depth at start, should fall within the range of 10 – 2000 meters.

date_f The finish date of the set must be a legitimate date.

Multiple column checks on station finish date, trip start date and trip finish date:
The station finish date must fall within the range of the trip start and finish dates.

time_f Station finish time must be a valid 24-hour time of between 0000 - 2359.

Multiple column checks on station start date/time and station finish date/time:
The station finish date/time must not be before the station start date/time. The finish date/start must be before the start date/time of any subsequent stations.

lat_f Latitude degree at finish, must be a valid latitude and degrees should fall within the range of 33 - 48 South, except for Bottom Longline vessels targeting toothfish species, that may fish down to 78 South.

long_f Longitude degree finish, must be a valid longitude and degrees should fall within the reasonable range of 164 East to 170 West.

ew_f Longitude East or West at finish, must be either "E" or "W".

Multiple column checks on station start and finish positions:
The start and finish positions should be within a defined maximum distance. The validation parameter for the distance between positions is set at 25 nautical miles. The time elapsed between the start and the finish of the station is taken into account on validation. The distance between stations must be within a distance that could be covered by the vessel in the elapsed time period between stations. The validation parameter is set at 15 knots for this check. Note, for drop lines, the end of the line set is not required as it is equal to the start position.

bottom_depth_f Bottom depth at finish, should fall within the range of 10 – 2000 meters.

fishing_method Fishing method is required for longline vessels, this must be a valid line fishing code, as listed in Appendix 1. Note this field is null for trawl vessels.

Observer trawl record (t_trawl)

Multiple column checks on trip and station number:
The combination of trip and station number must exist in the *t_station* table.

headline The headline height should fall within the reasonable range of 10 – 120 meters.

net_depth_s Net depth at start, should fall within the reasonable range of 10 – 2000 meters.

speed Speed should fall within the reasonable range of 1.0 – 6.0 knots.

net_depth_f Net depth at finish, should fall within the reasonable range of 10 – 2000 meters.

temp_surface Sea surface temperature should be in the range 8.0 to 24.0 degrees Celsius.

temp_headline Sea bottom temperature should be in the range 4.0 to 15.5 degrees Celsius.

Observer line record (t_line)

	Multiple column checks on trip and station number: The combination of trip and station number must exist in the *t_station* table.
topography	Bottom contour code must be a valid bottom type code as listed in Appendix 1.
no_hooks	The number of hooks should fall within the range of 10 – 15000.
bait1	Must be a valid species code as listed in the *curr_spp* table in the **rdb** database.
bait2	Must be a valid species code as listed in the *curr_spp* table in the **rdb** database.
percent_baited	Percent baited must be a value within the range 0> - <=100.
lf	Length frequency flag must be equal to "Y" or "N".
hooks_lost	The number of hooks lost must be a number greater than or equal to zero
	Multiple column checks on number of hooks set and number of hooks lost: The number of hooks lost must not exceed the number of hooks set.
assessment	Must be a valid catch assessment code as listed in Appendix 1.

Observer line catch record (t_catch)

	Multiple column checks on trip and set number: The combination of trip and set number must exist in the *v_line* table.
species	Must be a valid species code as listed in the *curr_spp* table in the **rdb** database.
discard	Must be a valid code, indicating discard status, as listed in Appendix 1.
catch_weight	Must be a number greater than zero.
no_fish	Must be a number greater than zero.
w_meth	Weight method code, consists of two parts; Part 1: a numeric code for the device used to weigh fish, Part 2: an Alpha to indicate method used to analyse the total catch. e.g. 1K means used Salter scales (1) and weighted in full (K). Each part must be a valid code as listed in Appendix 1, Line catch weight method codes.

Observer catch sampling record (t_general)

	Multiple column checks on trip and station number: The combination of trip and station number must exist in the *t_station* table.
species	Must be a valid species code as listed in the *curr_spp* table in the **rdb** database.
catch_weight	Must be a number greater than zero.
sample_weight	Must be a number greater than zero.
	Multiple column checks on catch and sample weights: Sample weight must be less than or equal to the catch weight.
sample_weigh_meth	Code for method used to weigh sample of fish, must be a valid weight code as listed in Appendix 1.
w_meth	Weight method code, must be a valid code combining two parts. Part 1: the location of the catch at the time of analysis. Part 2: an Alpha to indicate method used to analysis the total catch. e,g. means 7K analysis in processing area (7) and weighted in full (K). This code must compile the codes listed in Appendix 1.

Observer length frequency record (t_lth)

Multiple column checks on trip number, station number and species code:
The combination of trip number, station number and species code must exist in the *t_general* table.

meas_meth Must be a valid code as held in *t_fish_meas_codes* table in the **rdb** database.

lgth Must be a number greater than zero and should be a length within the range for the given species of fish, as held in the validation routines definition file.

no_m} Must be a valid integer 0 or greater.
no_f}
no_t} **Mulitple columns check on *no_t*, *no_m*, and *no_f*:**
The number in no_t must be equal to or not less than the sum of no_m and no_f for any given lgth.

no_f1} Must be a valid integer greater than or equal to 0.
no_f2}
no_f3}
no_f4}
no_f5}

Multiple column check on number of female gonad stages and the total number of females:
The sum of all staged females must not exceed the total females for a given length for that sample of fish.

Observer nonfish station record (t_nonfish_station)

	Multiple column checks on trip and tow number: The combination of trip and tow number must exist in the *t_station* table.
time_c	Time caught must be a valid 24-hour time of between 0000 - 2359.
lat_c	Must be a valid latitude and should fall within the reasonable range of 33 - 38 South.
long_c	Must be a valid longitude and should fall within the reasonable range of 164 East to 170 West.
ew_c	Longitude East or West caught, where recorded, must be either "E" or "W".
	Multiple column checks on time and position caught: The time and position caught (if known), should fall within the start & finish time / positions recorded for the station in the *t_station* record, for the given station.
wingspread	Distance between trawl wings should be between 20 - 300 meters.
depth_gear	Depth of gear should fall within the reasonable range of 10 – 2000 meters.
wind_knots	Must be a number greater than zero and should not exceed 0 - 70 knots.
wind_dir	Wind direction (degrees) must be in the range 0 to 359 from true north.
sea_state	Sea state on Beaufort scale, must be in the range 0 - 12, as listed in Appendix 1.
cloud	Cloud cover in eighths, must be in the range 0 to 8.
offal	Offal discarding code must a valid code as listed in Appendix 1.
tori_pole	Tori pole active code must be a valid code as listed in Appendix 1.
bird_device_o	Bird scaring device used code, must be “0” or “1”.
gear_event	Must be equal to “0” or “1”.
temp_surface	Sea surface temperature, should not exceed 23 degrees Celsius.
temp_headline	Sea temperature at headline, should not exceed 20 degrees Celsius.
tow_type	Must be a valid tow type code as listed in Appendix 1.
tow_config	Must be a valid tow configuration code as listed in Appendix 1.
tow_turns	The number of turns during the tow, should be in range 0 to 9.

Obs_lfs Nonfish Bycatch Record (t_nonfish_catch)

	Multiple column checks on trip number, tow number and time caught: The combination of trip number, tow number and time caught must exist in the *t_nonfish_station* table.
specimen_no	Number of the species in the tow, must be unique for this species within tow.
species} **species_obs}**	Must be a valid species code as listed in the *curr_spp* table in the **rdb** database.
lgth	Must be a number within a the range for the species, as listed in the non-fish lengths in Appendix 1.
girth	Must be a number greater than zero and should be in the range 60 – 1750.
blubber_mm	Must be a number greater than zero and should be in the range 5 – 80.
sex} **sex_obs}**	Must be a valid sex code (non-fish) as listed in Appendix 1.
alive	Must be a valid status code as listed in Appendix 1.
marked	Must be a valid marked code as listed in Appendix 1.
whole} **head}** **leg}** **ovary}** **stomach}** **teeth}** **skin}** **blubber}** **muscle}** **other}**	Must be equal to either a “0” or “1”.

Biological data for individual squid (t_fish_bio_asq)

	Multiple column checks on trip and station number: The combination of trip and station number must exist in the *t_station_squid* table.
species	Must be a valid squid species code as listed in the *curr_spp* table in the **rdb** database.
fish_no	Must be a number greater than zero and unique for the combination of trip_no, tow_no, species.
sex	Must be a valid sex code (non-fish) as listed in Appendix 1.
copul	Females only - Must be either 0 (not copulated) or 1 = (copulated).
lgth	Dorsal mantle length (DML) should be in the range 5 to 50 cm.

Technical specifications of squid jiggers (t_jig_specs)

fishing_yr	Fishing year, must be a combination of the last 2 digits of two consecutive years; e.g. 8788 (= Oct 87 to Sep 88).
lgth	Registered length of vessel should be within the range of 25 to 75 metres.
wt	Gross weight should be in the range of 70 to 999 tonnes.
speed_s	Service or normal speed should be in the range of 7 to 15 knots.
speed_m	Maximum speed should be in the range of 7 to 17 knots.
duration	Maximum duration at sea should be in the range 30 to 240 days.
gen1	Number of generators should be in the range of 1 to 4.
kva1	Power of "gen1" generators should be in the range 20 to 700 kva.
gen2	Number of generators should be in the range of 1 to 4.
kva2	Power of "gen2" generators should be in the range 20 to 700 kva.
gen3	Number of generators should be in the range of 1 to 4.
kva3	Power of "gen3" generators should be in the range 20 to 700 kva.
gen4	Number of generators should be in the range of 1 to 4.
kva4	Power of "gen4" generators should be in the range 20 to 700 kva.
hold_cap	Total fish hold capacity should be in the range 50 to 800 tonnes.
freezer	Capacity of quick or blast freezers should not exceed 95 tonnes/day.
jigs_h	Number of hand jig machines should be in the range 12 to 25.
jigs_m	Number of automatic jig machines should be in the range of 15 to 70.
lures	Number of lures per line should be in the range of 15 to 70.
dist_l	Distance between lures should not exceed 5 meters.
light1	Number of lights in light1 should not exceed 200.
watts1	Power of "light1" lights, should not exceed 5000 watts.

Light2	Number of lights in light2 should not exceed 200.
Watts2	Power of "light2" lights, should not exceed 5000 watts.
Light3	Number of lights in light3 should not exceed 100.
Watts3	Power of "light3" lights, should not exceed 5000 watts.
Light4	Number of lights in light4 should not exceed 100.
Watts4	Power of "light4" lights, should not exceed 5000 watts.

Seabird Warp-Strike Observations (Trawl) record (t_ warp_strike)

trip_no	Should be equal to a trip number held in the *t_trip* table.
station_no	Must be a unique integer within all station records, for a given trip number.
tow_date	The start date of the station must be a legitimate date. **Multiple column checks on station start date, trip start date and trip finish date**: The tow start date must fall within the range of the trip start and finish dates. The tow start date should be sequential between stations, for a given trip.
tow_start	Tow start time must be a valid 24-hour time of between 0000 - 2359.
tow_end	Tow end time must be a valid 24-hour time of between 0000 - 2359.
meal_plant	Meal plant on vessel, must be 'Y' or 'N'
meal_plant_on	Meal plant running during tow, must be 'Y' or 'N'

Seabird Warp-Strike Sampling Period record (t_ warp_strike_sample)

stn_key	stn_key must equal a stn_key held in the t_warp_strike table. **Multiple column checks on trip and station number:** The combination of trip and station number must exist in the *t_warp_strike* table.
time_start	Tow start time must be a valid 24-hour time of between 0000 - 2359.
time_end	Tow end time must be a valid 24-hour time of between 0000 - 2359.
large_range	Code for range large bird abundance must be between 0 – 3.
small_range	Code for range small bird abundance must be between 0 - 3.
tori_line	Tori line should be "Y", "N" or "X".
warp_scarer	Warp scarer should be "Y", "N" or "X".
bird_baffler	Bird baffler should be "Y", "N" or "X".
sprags_port	Sprags on portside warp must be "Y", "N" or "U".
sprags_starboard	Sprags on starboard warp must be "Y", "N" or "U".
grease	Grease on warps must be "P", "S", "B", or "N".
swell_dir	Swell direction should be between 1 - 12

wind_spd Wind speed on Beaufort scale should be between 0 -12

wind_dir Wind direction should be between 1 - 12

discharge_side Discharge side code must be a valid code as listed in Appendix 1.

discharge_rate Discharge rate code must be a valid code as listed in Appendix 1.

discharge_type Discharge type code should be a combination of valid codes as listed in Appendix 1.

Total Birds captured numbers for the tow record (t_bird_capture)

stn_key stn_key must equal a stn_key held in the t_warp_strike table.

recov_from Must be a valid code as listed in Appendix 1.

status Must be a valid code as listed in Appendix 1.

size Code for bird size, must be “L”, “S” or “N”.

Warp-strike mitigation devices (t_warp_strike_devices)

sample_key The sample_key must equal a sample_key held in the t_warp_strike_sample table.

d_complete Device complete code must be “Y”, “N” or “U “

deploy_sides Sides device deployed code must be “P”, “S”, “B” or “N”

Appendix 1 - Reference Code Tables

The information listed in this Appendix is current at the time of writing, and as implemented at November 30, 2001.

Origin codes

SOP	Scientific Observer Programme
ORM	Orange Roughy Management Company.
HMC	Hoki Management Company.
FRC	Fisheries Research Centre

Area codes

The valid area codes as in this list are a sub-set of the area codes in the **rdb** database.

AKE	East North Is. from North Cape to Bay of Plenty (FMA 1)
AKW	West North Is. from North Cp. to North Taranaki Bight (FMA 9)
CEE	East North Is. from south of Bay of Plenty to Wgtn (FMA 2)
CET	Challenger Plateau, beyond the EEZ (FMA)
CEW	West North Is. from South Taranaki Bight to Wgtn (FMA 8)
CHA	West Coast South Island to Fiordland incl. Kaikoura (FMA 7)
HOWE	Lord Howe Rise
KER	Kermadec (FMA 10)
LOUR	Louisville Ridge
PRET	Pukaki Rise ET - beyond the EEZ on the Pukaki Rise
SEC	East Coast South Island from Pegasus Bay to Catlins (FMA 3)
SOE	Chatham Rise (FMA 4)
SOI	Southern Offshore Islands - Auckland & Campbell Is. (FMA 6A)
SOU	South Island from Foveaux Strait to Fiordland (FMA 5)
SUB	Subantarctic incl. Bounty Is and Pukaki Rise (FMA 6)
TKET	Three Kings Rise, beyond the EEZ
TMAR	Tasmanian Ridge
WANB	Wanganella Bank
SOET	Southern Ocean (beyond the EEZ)

Line fishing method codes

The valid codes for the longline methods are a sub set of the overall gear method codes in the **rdb** database.

BLL	Bottom Longline
DAL	Drop or Dan Lines
TRO	Trolling lines
HAL	Handlines
TRL	Trot Lines

Squid fishing method codes
The valid codes for the squid methods are a sub set of the overall gear method codes in the **rdb** database.

01	bottom trawl
06	midwater trawl,
08	squid jigging.

Bottom contour codes

0	Unknown
1	Smooth / flat
2	Undulating
3	Hillocks
4	Rugged
5	Very rugged
6	Pinnacle
7	Canyon

Discard codes

R	Retained
D	Discarded
F	Finned
U	Unobserved
L	Lost
E	Eat
X	Not recorded / requested

Line catch weight method codes (for catch weight on *t_catch* records)
Part 1: the device used to weigh fish,

0	No scales used.
1	Salter scales (spring/manual)
2	SeaWay (motion compensated electronic) scales
3	Platform or Flatbed (manual) scales

Part 2:

A	Extrapolated from other catches (retrospectively).
B	Visual estimate
C	Inexact count *x* estimated average weight
D	Calculated by deduction (total minus other species)
E	Measured dimensions *x* density
F	Calculated from percentage composition
G	Calculated from percentage composition over several tows
H	Measuring fish and correlating length with weight
I	Accurate count *x* average weight for previous tows
J	Accurate count *x* average weight in random sample this tows
K	Weighed in full.

Sample weight method codes

1	Salter scales (spring/manual)
2	SeaWay (motion compensated electronic) scales
3	Platform or Flatbed (manual) scales
4	Accurate electronic scales (vessels)
99	Other weighing method used or weight estimated.

Trawl catch weight method codes (for catch weight for trawl methods.)
Part1: The location of the catch at the time of analysis.

1	In or spilling from codend.
3	Loose on deck.
5	In holding bins.
7	On sorting conveyor or in processing area
9	Packing area.

Part 2: Method used to analysis the total catch.

A	Extrapolated from other catches (retrospectively).
B	Visual estimate
C	Inexact count *x* estimated average weight
D	Calculated by deduction (total minus other species)
E	Measured dimensions *x* density
F	Calculated from percentage composition
G	Calculated from percentage composition over several tows
H	Measuring fish and correlating length with weight
I	Accurate count *x* average weight previous tows
J	Accurate count *x* average weight in random sample this tows
K	Weighed in full.
X	Any other technique (should be defined in comments).

Beaufort scale of wind force

0	Calm, glassy	< 1
1	Light air	1 - 3
2	Light Breeze	4 - 6
3	Gentle Breeze	7 - 10
4	Moderate Breeze	11 - 16
5	Fresh Breeze	17 - 21
6	Strong Breeze	22 - 27
7	Near Gale	28 - 33
8	Gale	34 - 40
9	Strong Gale	41 – 47
10	Storm	48 – 55
11	Violent Storm	56 – 63
12	Hurricane	64 +

Offal codes

0	Offal was not dumped overboard while shooting or hauling the gear.
1	Offal was dumped overboard while shooting the gear only.
2	Offal was dumped overboard while hauling the gear only.

3	Offal was dumped overboard while shooting and hauling the gear.
9	Offal undefined (pre trip numbers 780).

Tori pole used codes

0	No tori pole to CCAMLR specifications used.
1	Yes if a tori pole to CCAMLR specification used.

Tow type codes

1	Bottom throughout tow.
2	Midwater at relatively constant depth.
3	Midwater in a broad range of depths.
4	Mixed bottom & midwater.

Tow configuration codes

A	Straight line
B	”U”
C	Zigzag
D	Closed pattern (circle, loop etc)
E	Constant depth contour
F	Pinnacle fishing

Sex codes (for non-fish bycatch and squid)

0	Unsexed
1	Male
2	Female

Life status codes

1	Alive
2	Dead
3	Killed by crew
4	Dead – prior to catch, already decomposing

Marked codes

R	Retained.
D	Discarded unmarked.
M	Marked or tagged & discarded.

Non-fish length ranges

FUR	50 – 250 cm
POE	45 – 250
BDO	45 – 250
CDD	50 – 250
DDO	40 – 250
HDO	40 – 170
HSL	99 – 250
SEA	40 - 200

Catch assessment codes (for the degree of observation by the observer)

11	Observed all setting and hauling and the catch.
21	Observed the setting and the catch but not the hauling.
31	Observed the hauling and the catch but not the setting.
41	Observed the catch only, neither setting or hauling.
11	Observed nil, all figures from the crew
22	Observed setting only, neither hauling or catch.
32	Observed haul only, neither setting or catch.
22	Observed setting only, gear not retrieved (lost).
99	Observed parts of all operations.

Discharge side codes

P	Port
S	Starboard
B	Both
N	Neither / None

discharge_rate codes

0	none,
1	negligiable,
2	intermittent,
3	continueous

Discharge type codes

S	Sump water,
M	Minced,
C	cutter pump,
O	Offal,
D	Discards

Seabird warp-strike observations total birds "recovered from" codes

W	Warps
N	Net
M	Mitigation device
U	Unknown sources

Status codes for seabird warp-strike observations bird counts

D	Dead
I	Injured
A	Non injured
U	Unknown when no observation was made.

Codes for the Observer purse seine 'Vessel Activity Log'

ACTIVITY CODES	Using the codes listed in the right hand column of the Activity Log form, record the vessel's activity. Specify any details in the comments column (e.g. Activity 3; steaming to TAU, full load). Code explanations are provided below:
x	Prefix any activities not observed but noted by crew and subsequently transcribed with an "x".

1	Use each time the vessel commences a set (indicated by lowering the skiff off the vessel when a target school has been reached). The start time for Activity 1 should correspond to the "Start of Set" time on the CESD; the end time for Activity 1 should correspond to the "End of Set" time recorded on the CESD.
2	The vessel is searching for a school to target (e.g. using sonar or crows nest watch).
2a	When the vessel has been notified of a sighting and is traveling to the approximate location of the school.
3	If the vessel is traveling in to port, traveling out from port, or traveling from an overnight mooring / hove to their target fishing grounds (e.g. vessel may leave Tauranga and travel to Cape Brett).
4	If the vessel is unable to fish because of a vessel malfunction (most probably followed by Activity 3), or if (e.g.) a net has burst and has to be repaired prior to the next set.
5	If the vessel is unable to fish because of inclement weather (either in port or sheltering at sea).
5a	If the vessel is idle and waiting for the spotter plane to radio in a sighting
6a	You have boarded the vessel, but it is not yet ready to leave port (or is ready to leave but is unable to). Explain in comments
6	If, during your trip the vessel's holds become full and they come into port to offload their catch.
7	If the vessel feeds out the net (i.e. skiff off), with the sole intention of cleaning the net (i.e. they are *not* trying to catch anything, but are trying to remove debris etc that may have become entangled from the previous set).
8	If the vessel is investigating a school of fish (for example) to determine if it is suitable to target (eg. target species and school size).
11	If the vessel is moored/anchored overnight in a sheltered area (bay/inlet), or is drifting (hove to) overnight. Note lat/long and any other vessels in vicinity.
13a/b	If for any reason the vessel is unable to, or is not fishing, and no other "no fishing" codes are relevant. Note reason in comments field.
S1	The time that the spotter plane takes off (from airport) to search **A three letter code for the airport is recorded in the "port" field.**
S2	The time that the spotter plane lands
S1a	Record the time and the position of the school (lat/long) when the spotter pilot radios in a sighting to your vessel.
H1 / H2	Record the time that the helicopter takes off (from vessel) and returns to the vessel. Only relevant for larger vessels that carry a helicopter on board.
16O	Any other activity that is not covered by any of the codes listed (except "no fishing" - use 13a/b).

SCHOOL ASSOC	Using the codes on the right hand side of the page, record what the target/sighted school of fish were associated with. These fields indicate how the person who detected the school *initially* "spotted" it. Each field is explained below:
A1	If the pilot/skipper simply saw the school swimming beneath the surface (i.e. not stationary and feeding), with no birds present.
A2	If the pilot saw a "boil up" (i.e. localised sea surface turbulence), the school is probably feeding. Note in the comments section the likely species, eg krill.
A3	Often schools of pelagic species shelter from birds beneath flotsam (logs, dead cows, etc), if they are available. The spotter may radio to the skipper to investigate such debris.
A4	Vessels may deploy a Fish Aggregation Device (FAD), a raft, or a payao (usually in the Pacific when targeting tuna). An explanation of a payao is, "a big floating cylinder made of GI sheet four meters long and a meter wide. The crew put coconut fronds around the floating cylinder to provide shade for the fish. Naturally, the fish would gather around the payao". If the FAD is drifting freely, use A4, if anchored (fixed to one spot), use A5.
A5	FAD as above, if anchored (fixed to one spot).
A8	If a particular association is not listed, record "A8" and record in the comments space what the school was associated with (e.g. dolphins feeding).
A9	If the spotter / skipper saw birds feeding on the target school.

SCHOOL DETECT	Using the codes on the right hand side of the page, record who **initially** detected the target school.
D1	If someone on the vessel spotted the school **without assistance** from persons not on the vessel
D2	If the helicopter / airplane pilot radioed in a sighting.
D3	If another vessel / aircraft has spotted a school and (e.g. radioed your vessel to notify that) they deployed a beacon to mark that school, **and** the beacon is detected by your vessel.
D4	If your vessel is fitted with a bird radar device and this is used to detect birds feeding on a target school.
D5	If your vessel is fitted with a sonar and/or depth sounder and this is used to detect a target school.
D6	If another vessel has spotted a school and notified/radioed your vessel of that school and its location.
D8	If the detection method is not listed, record "D8" and explain in the comments field how the school was detected.

Codes for the Observer purse seine set.

Result

1 Entire school caught (on board)
2 Some caught / some lost
3 Skunked (entire school lost)
5 Caught unknown amount
6 Catch let go

Appendix 2 - Data entry, error checking, and loading

The data in ***obs_lfs*** have come predominately from the Scientific Observer Programme (SOP), while some data from various other sources is also included. The SOP trips began in 1986. In addition a small number of trips onboard commercial vessels, carried out by Fisheries Research Centre (FRC) staff, three earlier trips from 1979 to 1980, and several later trips are stored in the **obs_lfs** database. Other research providers under contract to the Ministry of Fisheries may supply data from industry observers. Data from other organizations are supplied in electronic form and are checked by their researchers working with the data as part of their contracts. These data are not all subject to the same level of checking by NIWA, as would be expected if NIWA was supplied with the raw data and was responsible for the data entry and checking of these data.

This section outlines the flow of paper-recorded data, for SOP data from collection through to its availability to researchers for analysis, and defines the separate tasks that are required to do this.

In this summary, the Observer data are recorded on hand written paper forms. Each trip is identified by its unique trip number, each tow or set by a sequential station number, each sample by a species. The date and time will also be recorded as part of the station data.

1. **Pre-key entry, visual checking and batching:**
At the completion of each trip the Observer should ensure that all pages are in order, and that all required data fields have been correctly filled out. The data are then forwarded via the Observer Programme, to a project team member, who checks the above, and forwards the data to key entry.

2. **Key entry of data:**
At this point, trained data entry operators key in the data from the collated forms to a electronic fixed format ASCII file format on computer by keyboard entry. NIWA uses the KEYS Data Emulator for data entry.

All data entry is verified, that is, each page of data are keyed in twice and the two results are crosschecked for mismatches. Any data entry operator errors are corrected at this point.

The electronic data files are transferred for error checking along with the original raw data file. At this point the data are now ready for error checking and formatting routines.

3. **Data error checking, validation, and grooming:**
Data files are put through a number of computer error checking (validation) routines that look for inaccuracies and inconsistencies within trips. Any errors detected are corrected. Data are then passed through these error-checking routines until the data reach a satisfactory standard that will allow them to be inserted in the appropriate database tables.

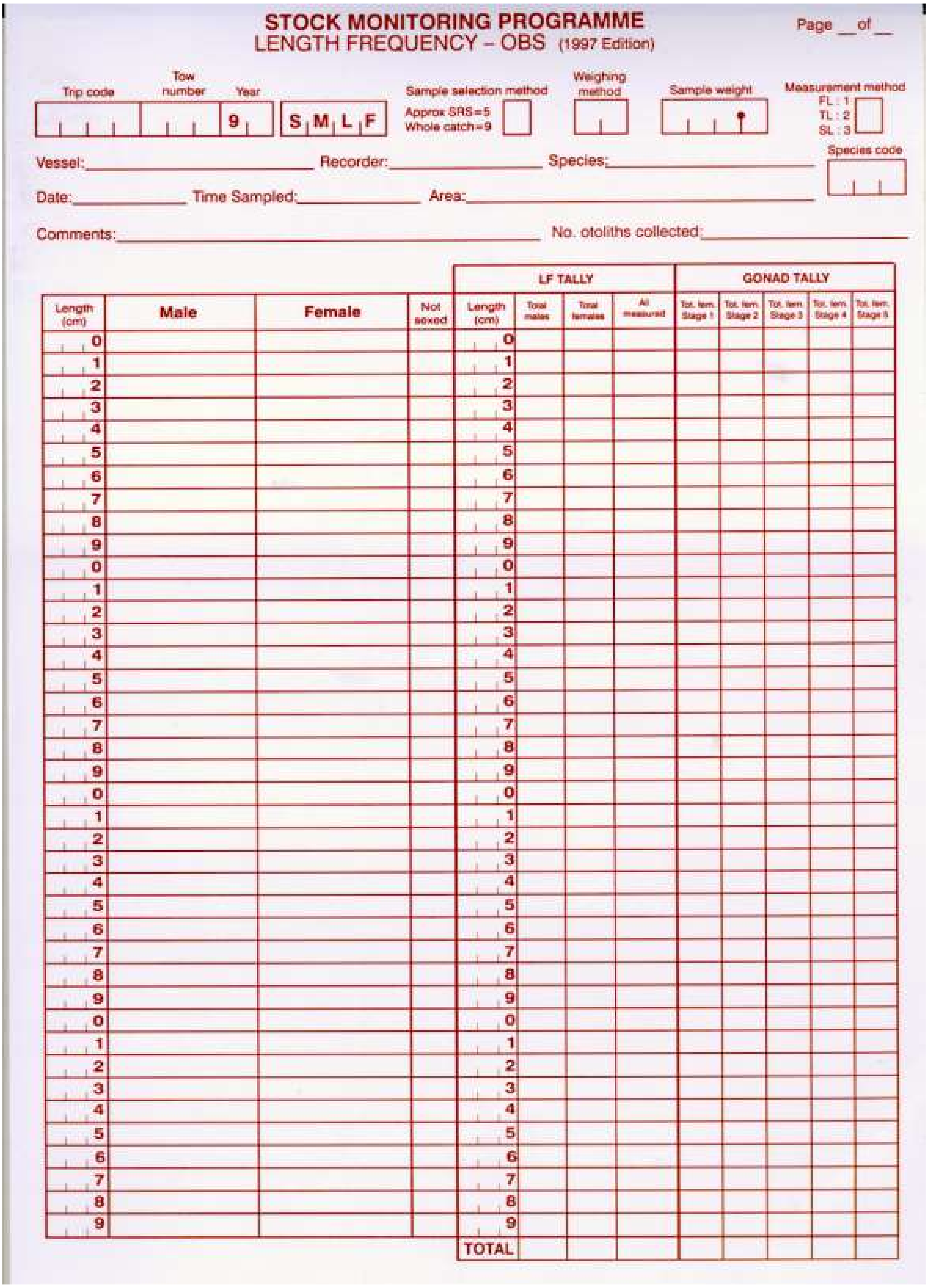
The data are inserted into “working tables”. This allows further checks of the integrity of the data, by taking advantage of relational databases ability to manipulate, match and compare related sets of data.

During this stage, data on the **obs** database from the catch and effort logbooks, are down loaded to the ***obs_lfs*** database working tables. Therefore, loading of the ***obs_lfs*** database is dependent upon the loading of correct data into the ***obs*** database.

4. **"Groomed", validated data loaded to database. Available for analysis:**
The clean, groomed, and validated data are inserted into the appropriate database (in this case **obs_lfs** on **snapper**) and now become available for extract and analysis.

The clean electronic data files and raw paper data are then archived for safekeeping.

Appendix 3 – Data forms



STOCK MONITORING PROGRAMME
LENGTH FREQUENCY – OBS (1997 Edition)

Page __ of __

Trip code	Tow number	Year		Sample selection method	Weighing method	Sample weight	Measurement method
		9	S M L F	Approx SRS=5 Whole catch=9			FL : 1 TL : 2 SL : 3

Vessel:________________ Recorder:________________ Species:________________ Species code

Date:__________ Time Sampled:__________ Area:________________

Comments:________________________ No. otoliths collected:__________

Length (cm)	Male	Female	Not sexed
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			

	LF TALLY			GONAD TALLY				
Length (cm)	Total males	Total females	All measured	Tot. fem. Stage 1	Tot. fem. Stage 2	Tot. fem. Stage 3	Tot. fem. Stage 4	Tot. fem. Stage 5
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
TOTAL								

Page ...... of ......

STOCK MONITORING PROGRAMME
MIDDLE DEPTH BIOLOGICAL DATA (1997 Edition)

Vessel .... Measurement method Recorder ....

Trip code Year 9 M D B D

FL : 1
TL : 2
SL : 3
Orbit : B

Species ....

Shot number	Fish number	Length (cm)	Sex	Stage	Otolith	Weighing method	Sample weight (kg)	Shot number	Fish number	Length (cm)	Sex	Stage	Otolith	Weighing method	Sample weight (kg)

Code	Sex	Female Gonad type	Otolith	SCI eggs use stage col.	SCI eggs use Otolith col.
0	—	—	No	—	Soft
1	Male	Resting	Yes	None	Hard
2	Female	Ripening	—	Blue	—
3	Unsexed	Mature	—	Orange	—
4	—	Running	—	Rose	—
5	—	Spent	—	—	—

Comments (area etc) ....

STOCK MONITORING PROGRAMME
NON-FISH BYCATCH (1995 edition)

Page __ of __

Trip code | Tow/Set number | S M N F | Date: _____

Vessel: _____ Observer: _____

If Known time and Position of catch

Time NZST	Latitude Deg	Min	S	Longitude Deg	Min	E/W	Gear: Wing spread (m)	Depth of gear (m)
			S					

Weather:

True wind speed knots	Wind direction ° true	Sea state beaufort	Cloud cover /8

Code	Sex	
0	Unsexed	No
1	Male	Yes
2	Female	–

Did a gear event likely to affect non-fish bycatch occur? If yes, record details below in comments. ☐

Was offal dumped while shooting gear? ☐

Was offal dumped while hauling gear? ☐

Was a tori pole to CCAMLR specifications deployed? ☐

Was another bird scaring device used? ☐ Specify: _____

Species code	Species number	Length (cm)	Sex	Tag number	Status on landing	Whole animal	Head	Leg	Ovary	Stomach	Teeth	Skin	Blubber	Muscle	Other, specify	Observed	Number seen

Comments: _____


Trip Number
Method
Target Species
FMA
Vessel
Observer
Page ..... of .....
Start of Set
End of Set
Set Number
Day
Month
Year
Date of Set
Time NZST
Latitude
Deg
Min
S
E/W
Longitude
Bottom Depth (m)
Topography
Number of Hooks Set
Bait 1
Bait 2
% Hooks Baited
Start of Haul
Day
Month
Year
Time NZST
Number of Hooks Lost
LF Y/N
Catch Assessment
Catch Details
Species name
Species code
Number of fish
Green weight (kg)
Weighing method
Comment 1:
Comment 2: