Database documentation: scallop

K. A. Mackay

NIWA Internal Report No. 71
2000

Contents

List of Figures

1 Database documentation 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 (MFish).

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 provides an introduction to the scallop survey database **scallop,** and is a part of the database documentation series produced by NIWA.

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 how all the tables fit in together, and their relationships to other databases.

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

2 Marine Research Scallop Surveys

Initially, surveys of scallop beds had been undertaken using diving methods. It was concluded from these surveys that the area sampled was too small to allow estimation of population size and that to increase this would involve extensive surveys requiring a large team of divers. Dredge surveys are seen as a more efficient means of estimating population abundance and length frequency distributions, as well as to determine scallop condition and to check for signs of large-scale mortality or recruitment.

Scallop dredge surveys have been carried out using two main methods:

1. *Fixed stations on a 1.0 nautical mile grid*: at each station two 5 minutes dredges are made in random compass directions. Double tows at each station are used as indicators of on-site variance. Scallops are tipped from the dredge onto the sorting tray and sub sampled by dividing the catch into two. One sub sample is accurately measured for the maximum length to nearest millimetre down. The remainder of the catch is counted and recorded as undersize (<100mm) or commercial length (>=100mm).
2. *Two phase stratified random dredges*: scallop beds are divided into strata based on their spatial extent. Dredge positions within strata are randomly selected, with about 10% of the dredges being allocated to the second phase. The dredge is towed in a direction and speed chosen by the skipper to optimise the performance of his gear until a distance of 0.5 nautical miles over the ground has been covered. At the completion of each tow, each dredge is retrieved and emptied onto a sorting tray on the boat. All live scallops are removed, counted, and the maximum length and/or height measured to the nearest millimetre down.

3 Data Structures

Initially, scallop surveys were formatted in the **scallop** database as tables suitable for input into the MultiPlot statistical programme. However, it soon became apparent that these data structures were not intuitive, nor were they robust enough to accept any future changes in scallop survey design. Therefore, **scallop**, was re-designed so that, at the conceptual level, it is very similar to **trawl**[1] database, but has attributes specific to scallop catches.

3.1 Table Relationships

This database contains several tables. The ERD for **scallop** (Figure 1) shows the physical data model structure[2] of the database and its entities (each entity is implemented as a database *table*) and relationships between these tables. Each table represents an object, event, or concept in the real world that has been represented in the database. Each *attribute* of a table is a defining property or quality of the table.

All of the table's attributes are shown in the ERD. The underlined attributes represent the table's primary key[3]. This schema is valid regardless of the database system chosen, and it can remain correct even if the Database Management System (DBMS) is changed.

Note that Figure 1 shows the main tables only. Most of the tables in the **scallop** database have some attributes, called foreign keys[4], which contain standard NIWA fisheries codes, such as *species.* These attributes provide links to tables in **scallop** and the **rdb** (research database) database, which contains the definitive list of standard codes. Therefore, an expanded ERD for these tables will follow (Figures 2 - 3).

Section 5 shows a listing of all the **scallop** tables as implemented by the Empress DBMS. As can be seen in the listing of the tables, a table's primary key has an unique index on it. Primary keys are generally listed using the format:

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

where the attribute(s) make up the primary key and the index name is the primary key name. Note that the typographical convention for the above (and subsequent) format is the square brackets [ ] may contain an item that is repeated zero or more times.

This unique index prevents records with duplicate key values from being inserted into the table, e.g., a new trip with an existing trip code, and hence ensures that every record can be uniquely identified.

---

[1] Mackay, K. 1998: Marine Research database documentation. 6. trawl. *NIWA Greta Point Internal Report No. 16.* 40p.

[2] Also known as a database *schema.*

[3] A primary key is an attribute or a combination of attributes that contains an unique value to identify that record.

[4] A foreign key is any attribute, or a combination of attributes, in a table that is a primary key of another table. Tables are linked together through foreign keys.

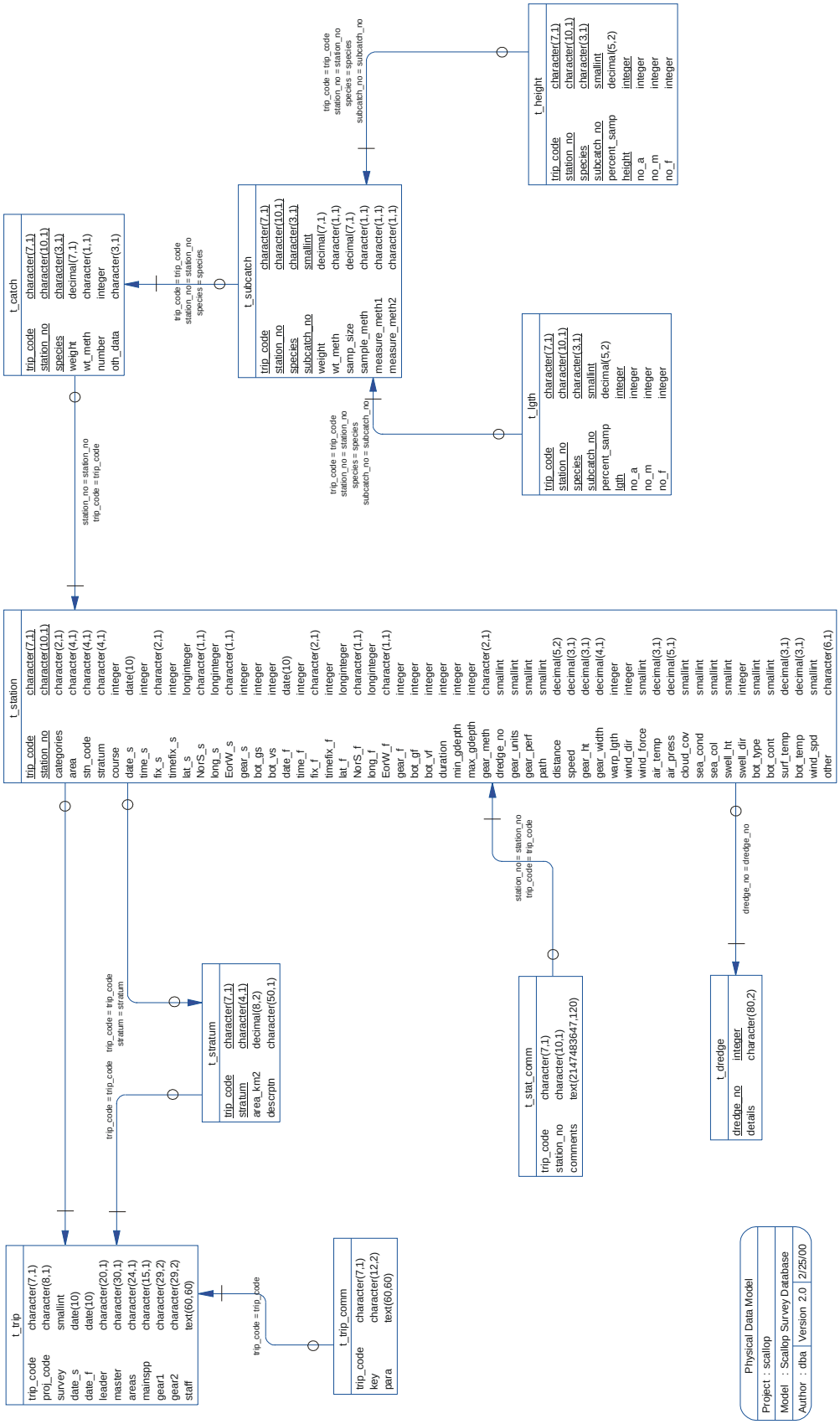

Figure 1: Entity Relationship Diagram (ERD) of the scallop database.

The **scallop** database is implemented as a relational database. That is, each table is a special case of a mathematical construct known as a ***relation*** and hence elementary relation theory is used to deal with the data within tables and their relationships between them. All relationships in **scallop** are of the type ***one-to-many***[5]. 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.

Every relationship has a mandatory or optional aspect to it. That is, if a relationship is mandatory, then it has to occur and least once, while an optional relationship might not occur at all. For example, in Figure 1, consider that relationship between the table ***t_trip*** and it's child table ***t_trip_comm.*** The symbol "O" by the child ***t_trip_comm*** means that a fish can have zero or many trip comments, while the bar by the parent ***t_trip*** means that for every trip comment there must be a matching trip.

Most of these tables contain foreign keys, which link these tables to each other and to tables in the **rdb** database (Figures 2 -3). The majority of these links are enforced by referential constraints[6]. 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: a parent record is deleted; the parent record is altered so that the relationship is lost; or a child record is entered without a parent record. Constraints are shown in the table listings by the following format:

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

For example, consider the following constraint found in the table ***t_station***:

```
Referential:      invalid trip code (trip_code) INSERT
                  t_trip (trip_code)
```

This means that the value of the attribute ***trip_code*** of a record upon insert into ***t_station*** must already exist in the parent table ***t_trip*** or the record will be rejected and an error message will be displayed.

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[, attribute])
```

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 for the index.

[5] A one-to-many relationship is where one record in a table (the ***parent***) relates to one or many records in another table (the ***child***).
[6] Also known as integrity checks.

3.2 Database design

As reflected by the ERD, the highest level of a scallop survey is a research survey. A survey may comprise of one or more trips as part of the same project. Details for each trip partaken within a survey are held in the table *t_trip* (Table 1). Each trip is uniquely identified by a trip code, stored as the attribute *trip_code*. Other details, such as the vessel name are also recorded in *t_trip*.

Note that the comments for a trip are held in a separate table *t_trip_comm* (Table 2). This means that one trip may have zero, one, or more than one comment associated with it. It can be argued that there is a one-to-one relationship between *t_trip* and *t_trip_comm*. After all, all comments to be made about one trip can be made in one comment. However, the trip comments have been separated from the trip details to two reasons:

I. Comments can be recorded at any time during a trip life-cycle. Rather than adding comments to those already recorded, it is easier just to create a new record.
II. To optimise query times, attributes with long field sizes, such as comments, are placed in separate tables to avoid being hit during tables scans for a regular expression.

For stratified scallop surveys, stratum details, such as stratum code and area (in square kilometres) are stored in the table *t_stratum* (Table 3). These strata may not necessarily be the same as those used for trawling during the same trip (as recorded in the **trawl** database). Notice that there is an optional link from *t_trip* to *t_stratum*; this means that not all trips have strata, i.e., they are unstratified scallop surveys.

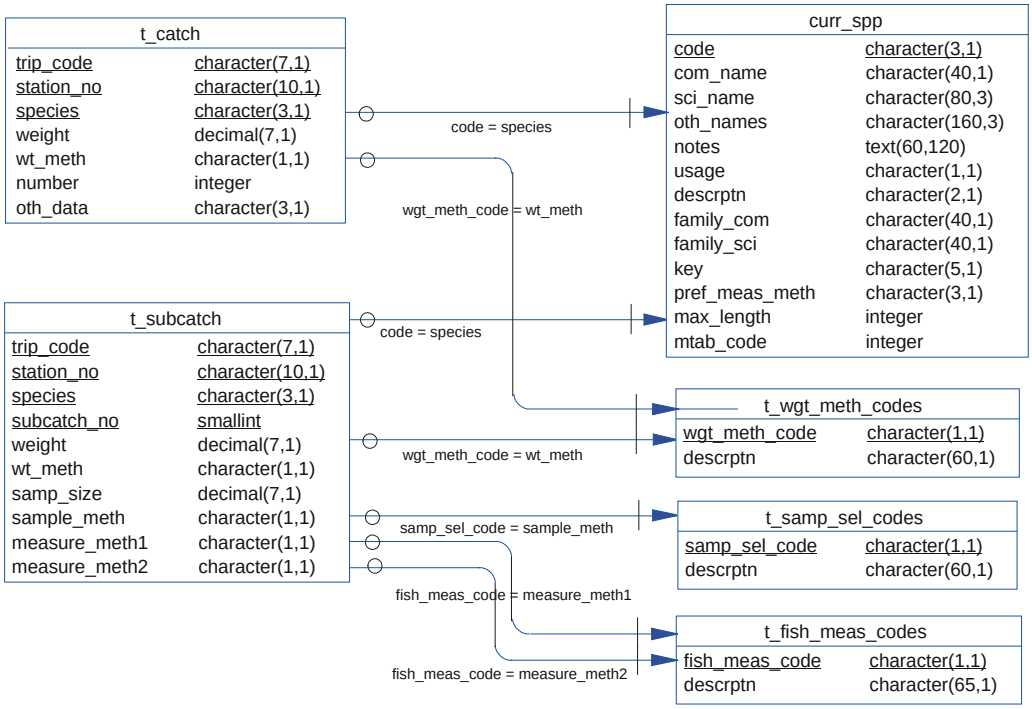

Figure 2: Expanded ERD for *t_catch* and *t_subcatch* showing the relationships to code tables in the rdb database

Any one trip also relates to many stations. This is a mandatory relationship: a trip has to have at least one station before it can be entered into the database. Generally, a station is the location at which the scallop gear was deployed. Details for the station, such as dredge start and finish location, time, depth, gear performance and environment parameters are stored in the table *t_station* (Table 4).

Note that a station may or may not occur within a pre-defined stratum (the table *t_station* contains the attribute *stratum*) and that one stratum may or may not contain stations. Therefore, there is a two-way optional many-to-one relationship between *t_station* and *t_stratum*.

Like the table *t_trip*, *t_station* has its own comments table *t_stat_comm* (Table 5). The same arguments that have been used for the creation of the *t_trip_comm* table also apply here.
Each station in a scallop dredge survey may produce a catch of several species. A catch from any one station is broken down into the different species, with each species being an individual record in the table *t_catch* (Table 6). Each record contains the species code, catch weight and other flags to indicate if a sample was taken for further measurement. The attributes *species* and *wt_meth* are codes that are foreign keys to tables in the **rdb** database (Figure 2) that provides explanations for the codes used. Not every station will produce a catch of shellfish, so again there is an optional one-to-many relationship between *t_station* and *t_catch.*

To cater for the instances where there are sub catches, the table *t_subcatch* (Table 7) stores information including sub catch weight, the method by which shellfish were selected for sampling from the sub catch, the weight of the shellfish used for sampling, and the shellfish measurement method used. Scallops commonly have two measurements taken for each shell: shell length and/or shell height. Hence there are dredge attributes in *t_subcatch* to record the measurement method: *measure_meth1* and *measure_meth2*. Each sub catch for a given trip, station and species is identified by the attribute *subcatch_no*.

Note that when sub catches are not used, the whole catch becomes one sub catch, and the attribute *subcatch_no* is equal to 1. Therefore, every record in *t_catch* has a one-to-many relationship to *t_subcatch.*

From a sub catch, a sample of shellfish may be taken for either shell length and/or shell height frequency measurements.

Shell length frequency data are stored in the table *t_lgth* (Table 8). Length class (millimetres) is stored at record level in this table, not individual fish. For each length class, the number of shellfish at that length is recorded. Counts for the number of males and females can also be recorded in *no_m* and *no_f* respectively. But scallops are usually not sexed so these attributes are left null. Note that the attribute *percent_samp* stores the percent of the sub catch that was sampled for length frequency, not the percent of the whole catch of the species.

Shell height frequency data are stored in the table *t_height* (Table 9). This table is conceptually identical to *t_lgth*, except that it recorded shell heights and not shell lengths.

Both *t_lgth* and *t_height* contain the foreign key species, which links them to the table *curr_spp* in the **rdb** database (Figure 3).

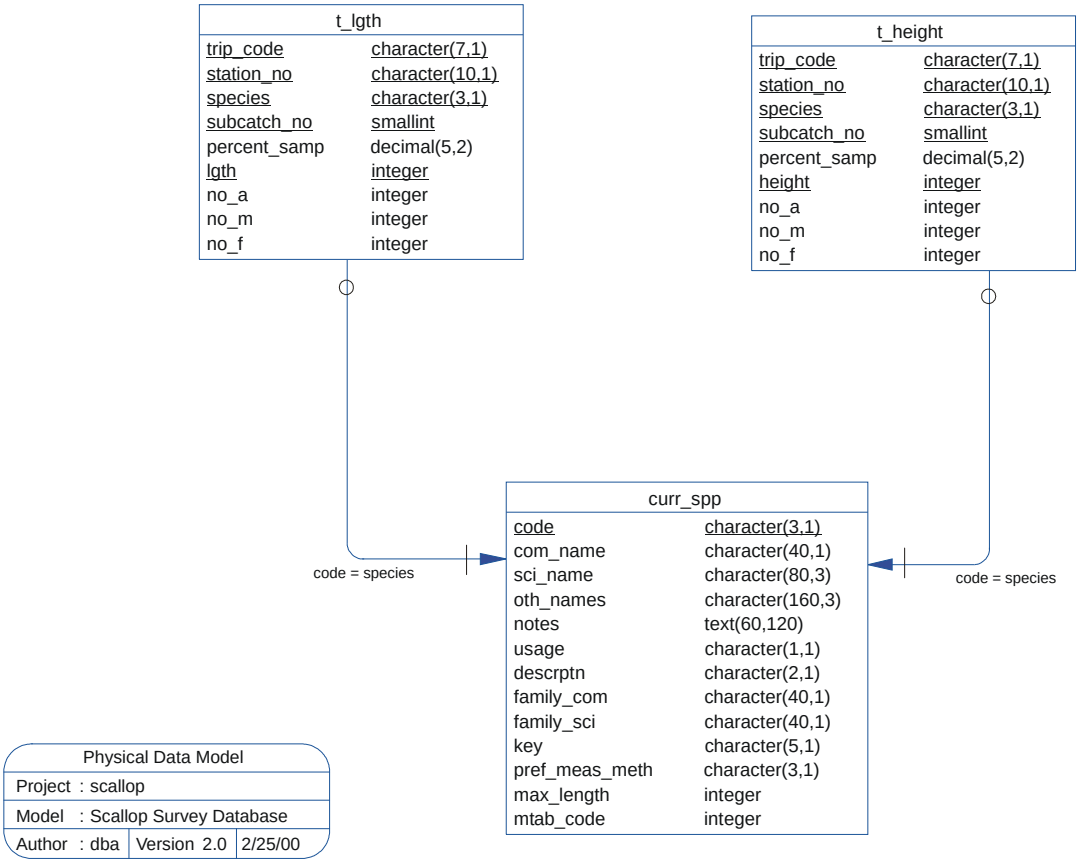

Figure 3: Expanded ERD of *t_lgth* and *t_height* showing relationships to the *curr_spp* table in the rdb database.

Future developments to the **scallop** database could see these two tables amalgamated into a generic *t_measurement*, which then would record any sort of measurement frequency for any sub catch.

Throughout the scallop database, vessels on which surveys were carried out on are referenced by a 3-character vessel code *vess_code*. Vessel names for these codes are recorded in the table *t_vessels* (Table 10).

Different types of scallop dredges used during surveys are referenced by a numeric dredge number *dredge_no*. Details for these dredges are recorded in the table *t_dredge* (Table 11).

4 Table Summaries

This database has eleven tables containing information pertaining to research scallop surveys. The following is a listing of the tables contained in the **scallop** database:

1. **t_trip :** contains profile information on all scallop trips.
2. **t_trip_comm :** contains comments for a particular trip.
3. **t_stratum :** contains details of strata surveyed for a trip.
4. **t_station :** contains data on location, gear used and environment at each station within a trip.
5. **t_stat_comm :** contains comments for a station in a trip.
6. **t_catch :** contains catch weight and abundance of species caught at a station
7. **t_subcatch :** contains weight and sample details of samples taken from the catch for further measurements.
8. **t_lgth :** contains shell length frequency data.
9. **t_height :** contains shell height frequency data.
10. **t_vessels:** contains vessel name and other general comments about vessels used during scallop trips.
11. **t_dredge :** contains details and comments about individual dredges used during a scallop trip.

5 scallop Tables

The following are listings of the tables in the **scallop** database, including attribute names, data types (and any range restrictions), and comments.

See Appendix 1 for attributes that have comments referring to the Trawl Instructions (unpub. NIWA report).

5.1 Table 1: t_trip

Comment: Profile information on all trips held in this database.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code - 3 char vessel name code, 2 digit year and 2 digit trip number.
proj_code	character(8,1)		Project or programme code for this trip.
survey	smallint		Survey id. (superceded by trip_code).
date_s	date(0)		Start date for the trip.
date_f	date(0)		Finish date for the trip.
leader	character(20,1)		Name of trip leader.
master	character(30,1)		Name of vessel master(s).
areas	character(24,1)		Codes of area(s) surveyed separated by commas (,)
mainspp	character(15,1) smatch “{[A-Z,]}”		Target species code(s) separated by commas.
gear1	character(29,2) match “{[0-9,.]}”		Gear code, gear method code, gear height, and gear width separated by commas for 1st gear used.
gear2	character(29,2) match “{[0-9,.]}”		Gear code, gear method code, gear height, and gear width separated by commas for 2nd gear used.
staff	text(20,60,20,1)		Name(s) of all staff on the trip.

Creator: dba

5.2 Table 2: t_trip_comm

Comment: Paragraphs explaining the trip, errors, extracts, etc.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code as defined in the trip table
key	character(12,1)		Keyword describing the topic of the comments.
para	text(61,256,80,1)		Any comments about this trip e.g. details about gear used apart from those recorded in the trip table

Creator: dba
Referential: invalid trip_code (trip_code) INSERT t_trip (trip_code)
Indices: NORMAL (2, 15) trip_comm_trip_code_ndx ON (trip_code)

5.3 Table 3: t_stratum

Comment: Table of strata surveyed in all trips.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code as in the trip table.
stratum	character(4,1)	No	Stratum code - unique within a trip
area_km2	decimal(8,2) > 0.00		Size of a stratum in sq. km. (km2) - must be greater than 0 sq. km.
descrptn	character(50,1)		Short description of the stratum e.g. location, depths

Creator: dba
Referential: invalid trip_code (trip_code) INSERT t_trip (trip_code)
Indices: NORMAL (2, 15) stra_stratum_ndx ON (stratum)
NORMAL (2, 15) stra_area_km2 ON (area_km2)
UNIQUE stra_key ON (trip_code, stratum)

5.4 Table 4: t_station

Comment: Data on location, gear used and environment at each station on a trip.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code as defined in the trip table
station_no	integer	No	Station number - unique within a trip
categories	character(2,1)		2 separate user-defined categories; definitions should be in trip comments
area	character(4,1)		Code describing area, refer to rdb:area_codes.
stn_code	character(4,1)		Code for a permanent station occupied repeatedly.
stratum	character(4,1)		Stratum number if trip is a stratified survey, else a transect code.
course	integer range 0-359		Course of vessel during the dredge (course-made-good).
date_s	date(5)		Starting date of the dredge (dd Mmm yy format).
time_s	integer range 0-2359		Starting time (24hr,NZST) of the dredge (hhmm format).
fix_s	character(2,1)		Method of fixing position at start of dredge, refer rdb:t_fix_meth_codes.
timefix_s	integer		Time (in minutes) elapsed since last position fix at the start of dredge.
lat_s	longinteger match “[3-6][0-9][0-5][0-9][0-9][0-9]”		Latitude of vessel at start of dredge (ddmmmm format, d=deg, m=min to 2 implied dec. pl.)
NorS_s	character(1,1) smatch “[NS]”		Dredge start position hemisphere.
long_s	longinteger match “1[7-8][0-9][0-5][0-9][0-9][0-9]”		Longitude of vessel at start of dredge (dddmmmm format, d=deg, m=min to 2 implied dec. pl.)
EorW_s	character(1,1) smatch “[EW]”		Dredge start position meridian.

Attributes	Data Type	Null?	Comment
gear_s	integer		Depth (m) of lowest part of gear (groundrope) at the start of dredge.
bot_gs	integer		Depth (m) of sea bottom at gear position at start of the dredge.
bot_vs	integer		Depth (m) of sea bottom at vessel position at start of the dredge.
date_f	date(5)		Finishing date of the dredge (dd Mmm yy format).
time_f	integer range 0-2359		Finishing time (24hr,NZST) of dredge (hhmm format).
fix_f	character(2,1)		Method of fixing position at end of dredge, refer rdb: t_fix_meth_codes.
timefix_f	integer		Time (in minutes) elapsed since last position fix at end of the dredge.
lat_f	longinteger match “[3-6][0-9][0-5][0-9][0-9][0-9]”		Latitude of vessel at end of dredge (ddmmmm format, d=deg, m=min to 2 implied dec. pl.)
NorS_f	character(1,1) smatch “[NS]”		Dredge finish position hemisphere.
long_f	longinteger match “1[7-8][0-9][0-5][0-9][0-9][0-9]”		Longitude of vessel at end of dredge (dddmmmm format, d=deg, m=min to 2 implied dec. pl.)
EorW_f	character(1,1) smatch “[EW]”		Dredge finish position meridian.
gear_f	integer		Depth (m) of lowest part of gear (groundrope) at end of the dredge.
bot_gf	integer		Depth (m) of sea bottom at gear position at end of dredge.
bot_vf	integer		Depth (m) of sea bottom at vessel position at end of dredge.
duration	integer		Dredge duration (minutes).
min_gdepth	integer		Minimum depth (m) of lowest part of gear (groundrope) during the dredge.
max_gdepth	integer		Maximum depth (m) of lowest part of gear (groundrope) during the dredge.
gear_meth	character(2,1)		Gear method code, descriptions in rdb:meth_codes.
dredge_no	smallint		Code for set of gear used, details in t_dredge.

Attributes	Data Type	Null?	Comment
gear_units	smallint		Number of units of gear used in the dredge.
gear_perf	smallint range 1-4		Code for performance of gear during the dredge, refer to the trawl instructions.
path	smallint range 1-8		Code describing configuration of path of dredge, refer to the trawl instructions.
speed	decimal(3,1)		Average speed through water during dredge (knots).
distance	decimal(4,2)		Distance of gear over bottom (nautical miles).
gear_ht	decimal(3,1)		Average height (m) of dredge.
gear_width	decimal(4,1)		Average width (m) of dredge.
warp_lgth	integer		Length of warp during the dredge (m).
wind_dir	integer range 0-359, 999		Wind direction (degrees true), 999=No wind.
wind_force	smallint range 0-12		Wind force on Beaufort scale.
air_temp	decimal(3,1)		Air temperature (degrees C).
air_press	decimal(5,1)		Air pressure (millibars).
cloud_cov	smallint range 0-8		Code describing cloud cover during dredge, refer to trawl instructions.
sea_cond	smallint range 0-9		Code describing condition of sea, refer trawl instructions.
sea_col	smallint range 1-8		Code describing colour of sea, refer trawl instructions.
swell_ht	smallint range 1-3		Code describing height of swell, refer trawl instructions.
swell_dir	integer range 0-359, 999		Direction of the swell (degrees true).

Attributes	Data Type	Null?	Comment
bot_type	smallint range 0-9		Code describing sea bottom type, refer trawl instructions.
bot_cont	smallint range 0-5		Code describing sea bottom contour, refer trawl instructions.
surf_temp	decimal(3,1)		Surface temperature (degrees C).
bot_temp	decimal(3,1)		Temperature at bottom (degrees C).
wind_spd	smallint		Wind speed from anemometer (m/s) (1knot=0.51m/s).
other	character(6,1)		Any other details, should be fully commented.

Creator: dba

Referential:
invalid trip_code (trip_code) INSERT t_trip (trip_code)
invalid area code (area) INSERT rdb : area_codes (code)
invalid fix_s code (fix_s) INSERT rdb : t_fix_meth_codes (fix_meth_code)
invalid fix_f code (fix_f) INSERT rdb : t_fix_meth_codes (fix_meth_code)
invalid gear code (gear_meth) INSERT rdb : meth_codes (code)

Indices:
UNIQUE stat_key ON (trip_code, station_no)
NORMAL (2, 15) stat_max_gdepth_ndx ON (max_gdepth)
NORMAL (2, 15) stat_min_gdepth_ndx ON (min_gdepth)
NORMAL (2, 15) stat_station_no_ndx ON (station_no)
NORMAL (2, 15) stat_gear_meth_ndx ON (gear_meth)

5.5 Table 5: t_stat_comm

Comment: Comments for a station in a trip.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code as in the trip table
station_no	integer	No	Station number as in station table
comments	text(60,120,60,1)	No	Comments for this station – should include comments about catch & LF data or any special action taken during dredge

Creator: dba
Referential: invalid trip_code, station_no (trip_code, station_no)
INSERT t_station (trip_code, station_no)
Indices: NORMAL (2, 15) scom_trip_code_ndx ON (trip_code)
NORMAL (2, 15) scom_station_no_ndx ON (station_no)

5.6 Table 6: t_catch

Comment: Information (weight, number caught etc) on all species caught at each station on a trip.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code as in the trip table
station_no	integer	No	Station number as in station table
species	character(3,1)	No	Species code, refer to rdb:curr_spp.
weight	decimal(7,1)		Weight (kg) of the species caught at that station.
wt_meth	character(1,1)		Code of method used to determine weight of catch, refer rdb:t_wgt_meth_codes.
number	integer		Counted or estimated number of this species.
oth_data	character(3,1) match “\{[01]\}”		Col. 1=L/F?, Col. 2=Biologicals, Col. 3=Otoliths. In each column, 1=Yes & 0 or blank=No.

Creator: dba
Referential: invalid trip_code, station_no (trip_code, station_no) INSERT t_station (trip_code, station_no)
invalid species (species) INSERT rdb : curr_spp (code)
invalid wt_meth code (wt_meth) INSERT rdb : t_wgt_meth_codes (wgt_meth_code)
Indices: NORMAL (2, 15) ctch_station_no_ndx ON (station_no)
NORMAL (2, 15) ctch_species_ndx ON (species)
NORMAL (2, 15) ctch_trip_code_ndx ON (trip_code)

5.7 Table 7: t_subcatch

Comment: Information (weight, sample weight etc) on each sub catch for each species. Generally, the sub catch is identical to the whole catch for any species.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code as in the trip table
station_no	integer	No	Station number as in station table
species	character(3,1)	No	Species code, refer to rdb:curr_spp
subcatch_no	smallint	No	Sequential number to identify each sub catch of a species taken from the whole catch for that species.
weight	decimal(7,1)		Weight (kg) of the species caught at that station.
wt_meth	smallint		Code of method used to determine weight of catch, refer rdb:t_wgt_meth_codes.
samp_size	decimal(7,1)		Size (either as weight (kg) or numbers) of the sample taken for measuring.
sample_meth	character(1,1)		Code of method used in sampling LFs (if done), refer rdb:t_samp_sel_codes.
measure_meth1	character(1,1)		Code of 1st method used to measure fish lengths (if LFs done), refer rdb:t_fish_meas_codes.
measure_meth2	character(1,1)		Code of 2nd method used to measure fish lengths (if LFs done), refer rdb:t_fish_meas_codes.

Creator: dba
Referential: no such catch (trip_code, station_no, species) INSERT t_catch (trip_code, station_no, species)
invalid wt_meth code (wt_meth) INSERT rdb : t_wgt_meth_codes (wgt_meth_code)
invalid samp meth code (sample_meth) INSERT rdb : t_samp_sel_codes (samp_sel_code)
invalid meas meth code (measure_meth1) INSERT rdb : t_fish_meas_codes (fish_meas_code)
invalid meas meth code (measure_meth2) INSERT rdb : t_fish_meas_codes (fish_meas_code)
Indices: NORMAL (2, 15) subc_trip_code_ndx ON (trip_code)
NORMAL (2, 15) subc_station_no_ndx ON (station_no)
NORMAL (2, 15) subc_species_ndx ON (species)

5.8 Table 8: t_lgth

Comment: Length frequency data on sampled species in a trip.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code as in the trip table
station_no	integer	No	Station number as in station table
species	character(3,1)	No	Species code, refer to rdb:curr_spp.
subcatch_no	smallint	No	Subcatch number as in sub catch table.
lgth	integer	No	Measured length (mm) of the shellfish.
percent_samp	decimal(5,2) range 0-100		Sampling percentage associated with this record.
no_a	integer		Number of all measured shellfish at this length in this sub catch.
no_m	integer		Number of all measured male shellfish at this length in this sub catch.
no_f	integer		Number of all measured female shellfish at this length in this sub catch

Creator: dba
Referential: invalid trip_code, station_no (trip_code, station_no) INSERT t_station (trip_code, station_no)
invalid species (species) INSERT rdb : curr_spp (code)
Indices: NORMAL (2, 15) lgth_trip_code_ndx ON (trip_code)
NORMAL (2, 15) lgth_station_no_ndx ON (station_no)
NORMAL (2, 15) lgth_species_ndx ON (species)

5.9 Table 9: t_height

Comment: Shell height frequency data on sampled species in a trip.

Attributes	Data Type	Null?	Comment
trip_code	character(7,1)	No	Trip code as in the trip table
station_no	integer	No	Station number as in station table
species	character(3,1)	No	Species code, refer to rdb:curr_spp.
subcatch_no	smallint	No	Subcatch number as in sub catch table.
height	integer	No	Measured height (mm) of the shellfish.
percent_samp	decimal(5,2) range 0-100		Sampling percentage associated with this record.
no_a	integer		Number of all measured shellfish at this height in this sub catch.
no_m	integer		Number of all measured male shellfish at this height in this sub catch.
no_f	integer		Number of all measured female shellfish at this height in this sub catch

Creator: dba
Referential: invalid trip_code, station_no (trip_code, station_no) INSERT t_station (trip_code, station_no)
invalid species (species) INSERT rdb : curr_spp (code)
Indices: NORMAL (2, 15) height_trip_code_ndx ON (trip_code)
NORMAL (2, 15) height_station_no_ndx ON (station_no)
NORMAL (2, 15) height_species_ndx ON (species)

5.10 Table 10: t_vessels

Comment: Details of vessels and vessel codes used for "trip_code".

Attributes	Data Type	Null?	Comment
vess_code	character(3,1)	No	3 char code for vessel as used in "trip_code"
vess_name	character(20,1)	No	Vessel name.
comments	text(20,20,20,1)		General comments about the vessel.

Creator: dba
Indices: NORMAL (2, 15) BTREE vessels_vess_code ON (vess_code)

5.11 Table 11: t_dredge

Comment: Details of each dredge used.

Attributes	Data Type	Null?	Comment
dredge_no	integer	No	Unique number for each dredge.
details	character(80,2)		Details about the dredge.

Creator: dba
Indices: NORMAL (2, 15) BTREE dredge_dredge_no ON (dredge_no)

6 scallop business rules

6.1 Introduction to business rules

The following are a list of business rules pertaining to the **scallop** database. A business rule is a written statement specifying what the information system (i.e., any system that is designed to handle scallop dredge survey data) must do or how it must be structured.

There are three recognized 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. Referential constraints, range checks, and algorithms both in the database and during data validation implement the formula and validation type rules.

6.2 Summary of rules

Scallop survey trip details (t_trip)

trip_code Trip code, must be unique. Trip codes are in the following format: 3 character vessel code (see the *t_vessels* table for available codes); 2 digit year (e.g., 99 = 1999, 00 = 2000); 2 digit sequential trip number for each vessel each year.

proj_code Project code must be a valid code within the NIWA project management system.

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

date_f The start date of the trip must be a legitimate date.

Multiple column checks on date:
The start date must not be later than the finish date.

areas Each of the listed area codes must be a valid code as listed in the *area_codes* table in the **rdb** database.

mainspp Each of the listed species codes must be a valid code as listed in the *curr_spp* table in the **rdb** database.

gear1 – **gear2** Gear descriptions. The following describe the format, and where applicable, the business rules for the description of gear used during a trip:

- **gear number** Must be a unique, sequential number from 1 to 6 to identify each unit of gear.
- **gear method** Must be a valid code as listed in the *meth_codes* table in the **rdb** database.
- **gear height**
- **gear width**

Scallop survey trip comments (t_trip_comm)

trip_code Must be equal to a trip code as listed in the *t_trip* table.

Scallop survey stratum details (t_stratum)

trip_code Must be equal to a trip code as listed in the *t_trip* table.

Scallop survey station details (t_station)

trip_code	Must be equal to a trip code as listed in the *t_trip* table.
station_no	Must be a unique number within a single trip.
area	Area code must be a valid code as listed in the *area_codes* table in the **rdb** database.
course	Course must be within the range of 0 – 359 degrees.
date_s	The date at the start of a station must be a legitimate date. **Multiple column checks on start date:** The date must fall within the range of the range of the trip start and finish dates.
time_s	Start time of the station must be a valid 24-hour time and fall within the range of 0 – 2359 hours.
fix_s } **fix_f }**	The method of position fix code must be valid code as listed in the *t_fix_meth_codes* table in the **rdb** database.
lat_s	Must be a valid latitude
NorS_s	Northern or Southern Hemisphere at station start, must be equal to either "N" or "S".
long_s	Must be a valid longitude.
EorW_s	Longitude east or west at station start, must be equal to either "E" or "W".
bot_gs	Depth of sea bottom must not be less than depth of gear
date_f	The date at the finish of a station must be a legitimate date. **Multiple column checks on finish date:** The date must fall within the range of the range of the trip start and finish dates.
time_f	Finish time of the station must be a valid 24-hour time and fall within the range of 0 – 2359. **Multiple columns checks on date and time**: The start date must not be later than the finish date and within a reasonable time period.
lat_f	Must be a valid latitude

NorS_f	Northern or Southern Hemisphere at station finish, must be equal to either "N" or "S".
long_f	Must be a valid longitude.
EorW_f	Longitude east or west at station finish, must be equal to either "E" or "W".
	Multiple columns checks on position: The finish position should be within a reasonable distance from the start position for the gear type used.
bot_gf	Depth of sea bottom must not be less than depth of gear
duration	Duration should be within the reasonable range of 0 – 200 minutes
min_gdepth	Minimum gear depth must be less than or equal to the depth of gear at the start and finish of the station.
max_gdepth	Maximum gear depth must be greater than or equal to the minimum gear depth and the depth of gear at the start and finish of the station
gear_meth	Gear method code must be a valid code as listed in the *meth_codes* table in the **rdb** database.
dredge_no	Dredge number must be a valid code as listed in the *t_dredge* table.
gear_code	Must within the range 1 – 6 to relate to gear details in *gear1* to *gear6* respectively in the *t_trip* table.
gear_perf	The gear performance code must be valid code as listed in Appendix 1.
path	The path code must be valid code as listed in Appendix 1.
speed	The vessel's recorded speed during the station should be within the range 0 – 5 knots and be reasonable for the gear method.
distance	The distance traveled during the station should be reasonable for the gear] method.
	Multiple columns check on: distance; start and finish positions; and speed and start/finish times: The distance traveled during a station as calculated by (1) the difference between start and finish positions; (2) speed * elapsed time; and (3) recorded distance should be in approximate agreement.
wind_dir	Wind direction must fall within the range of 0-359, 999.

wind_force	Wind force must fall within the range of 0 – 12.
air_temp	Air temperature should fall within the reasonable range of 5 – 30.
air_press	Air pressure should fall within the reasonable range of 960 to 1040.
cloud_cov	Cloud cover must fall within the range of 0-8.
sea_cond	The sea condition code must be valid code as listed in Appendix 1.
sea_col	The sea colour code must be valid code as listed in Appendix 1.
swell_ht	The swell height code must be valid code as listed in Appendix 1.
swell_dir	Wind direction must fall within the range of 0-359, 999.
bot_type	The bottom type code must be valid code as listed in Appendix 1.
bot_cont	The bottom contour code must be valid code as listed in Appendix 1.
surf_temp	Sea surface temperature should fall within the reasonable range of 5 – 28.
bot_temp	Sea bottom temperature should fall within the reasonable range of 3 – 25.
wind_spd	Wind speed should fall within the reasonable range of 0 - 30.

Scallop survey station comments (t_stat_comm)

trip_code Must be equal to a trip code as listed in the *t_trip* table.

station_no Must be a unique number within a single trip.

Multiple columns check on trip code and station number:
The combination of trip code and station number must exist in the *t_station* table.

Scallop survey catch details (t_catch)

trip_code Must be equal to a trip code as listed in the *t_trip* table.

station_no Must be a unique number within a single trip.

Multiple columns check on trip code and station number:
The combination of trip code 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.

weight Must be a valid number greater then 0

wt_meth Must be a valid code as listed in the *t_wgt_meth_codes* table in the **rdb** database.

oth_data Must be up to 3 characters long, with each character being a “1” (meaning presence), “0” (meaning absence), or “ ” (meaning not recorded).

Scallop survey sub catch details (t_subcatch)

trip_code	Must be equal to a trip code as listed in the *t_trip* table.
station_no	Must be a unique number within a single trip.
species	Must be a valid species code as listed in the *curr_spp* table in the **rdb** database. **Multiple columns check on trip code, station number, and species:** The combination of trip code, station number, and species must exist in the *t_catch* table.
subcatch_no	Must be a unique number within a single trip code, station number, and species.
weight	Must be a valid number greater then 0
wt_meth	Must be a valid code as listed in the *t_wgt_meth_codes* table in the **rdb** database.
sample_meth	Must be a valid sample selection method code as listed in the *t_samp_sel_codes* table in the **rdb** database.
measure_meth1	Must be a valid fish measurement method code as listed in the *t_fish_meas_codes* table in the **rdb** database. **Multiple columns check on species and measure_meth:** The fish measurement method code must be valid for the species sampled.
measure_meth2	Must be a valid fish measurement method code as listed in the *t_fish_meas_codes* table in the **rdb** database. **Multiple columns check on species and measure_meth:** The fish measurement method code must be valid for the species sampled.

Scallop survey length frequency details (t_lgth)

	Multiple columns check on trip code, station number, species, and sub catch number: The combination of trip code, station number, species, and sub catch number must exist in the *t_subcatch* table.
species	Must be a valid species code as listed in the *curr_spp* table in the **rdb** database.
lgth	Should be within the reasonable range of 5 - 200
	Multiple columns check on species and length: The fish length should be less than the maximum-recorded fish length for the species as recorded in the *curr_spp* table in the **rdb** database.
percent_samp	Must be a valid percentage up to 100%
	Multiple columns check on percentage sampled and *t_subcatch:sample_meth*: The sample selection method code must valid with the percentage sampled
no_m} **no_f}** **no_a}**	Must be a valid integer greater than 0 **Mulitple columns check on *no_a, no_m,* and *no_f*:** The number in *no_a* must be equal to or less than the sum of *no_m* and *no_f*.

Scallop survey shell height frequency details (t_height)

	Multiple columns check on trip code, station number, species, and sub catch number: The combination of trip code, station number, species, and sub catch number must exist in the *t_subcatch* table.
species	Must be a valid species code as listed in the *curr_spp* table in the **rdb** database.
height	Should be within the reasonable range of 5 - 200
percent_samp	Must be a valid percentage up to 100%
	Multiple columns check on percentage sampled and *t_subcatch:sample_meth*: The sample selection method code must valid with the percentage sampled
no_m} **no_f}** **no_a}**	Must be a valid integer greater than 0 **Mulitple columns check on *no_a, no_m,* and *no_f*:** The number in *no_a* must be equal to or less than the sum of *no_m* and *no_f*.

Scallop survey vessel codes (t_vessels)

vess_code	Must be a unique value.
vess_name	Must contain a value.

Scallop survey dredge codes (t_dredge)

dredge	Must be a unique value.

Appendix 1 – Reference Code Tables

Gear performance code

1. Excellent
2. Satisfactory, catch unlikely to be reduced by performance
3. Unsatisfactory, catch probably reduced by malfunction or damage
4. Unsatisfactory, catch reduced by malfunction or damage

Path code

1. Horizontal straight line
2. Vertical straight line
3. Closed circle or loop
4. Closed triangle or square
5. Zigzag
6. U-bend
7. Contour at constant depth
8. Retrack on straight line

Sea condition code

0	Calm, glassy	0m
1	Calm	0 – 0.1m
2	Smooth	0.1 – 0.5m
3	Slight	0.5 – 1m
4	Moderate	1 – 2.5m
5	Rough	2.5 – 4m
6	Very rough	4 – 6m
7	High	6 - 10m
8	Very high	10 – 15m
9	Huge	over 15m

Sea colour code

01	Deep blue
02	Blue
03	Light blue
04	Greeny blue
05	Bluey green
06	Deep green
07	Green
08	Yellow green

Swell height code

1	Low	0 – 2m
2	Moderate	2 – 4m
3	Heavy	over 4m

Bottom contour code

0	Unknown
1	Smooth/flat
2	Undulating
3	Hillocky
4	Rugged
5	Very rugged

Bottom type code

0	Unknown
1	Mud or ooze
2	Mud with some sand
3	Sand
4	Sand/gravel and shells
5	Shells (broken)
6	Gravel
7	Rock
8	Coral
9	Stone
10	Live shell beds
11	Mud with broken shells
12	Sponge beds