Database documentation: tuna

F. Wei

NIWA Internal Report

Last modified in April 2002

Contents

List of Figures/Tables

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 database **tuna**, and is a part of the database documentation series produced by NIWA. It supersedes the previous documentation by H.A Dean (1998) on this database.

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 and business rules. 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 **tuna** database.

2 Commercial tuna data

2.1 Data sources

2.1.1 New Zealand Data

The data in **tuna** are derived from Ministry of Fisheries catch and effort data and cover a period from 1 January 1980 to the present. Data compiled from before 1988 are sourced from the Fisheries Statistics Unit (FSU), a precursor to the Ministry of Fisheries catch and effort database. Recent data are derived from the Tuna Longlining Catch and Effort Return (TLCER) and Catch and Effort Landing Return (CELR) forms.

Some scientific research data are also collected by commercial operators for the Real Time Monitoring Programme (RTMP), using the RTMP form, which collects single fish length and weight records.

Nearly all the commercially collected data can be linked to the Scientific Observer longline tuna database, **l_line,** which holds hook, catch, and environment data collected by New Zealand observers.

2.1.2 Australian Data

A series of independent tables hold data collected by Australian based observers for the Real Time Monitoring Programme (RTMP). (*These data have more detail than New Zealand commercial data and should ideally be held in **l_line***.)

2.1.3 Japanese Data

Data summaries provided by the Japanese are held in accordance with exchange protocols established by the Commission for the Conservation of the Southern Bluefin Tuna (CCSBT). These data are in the standard Yellow Book summary format, which summarises catch and effort by five-degree squares of latitude and longitude and monthly periods of time.

3 Data structures

3.1 Introduction

The **tuna** database began as a collection of surface longlining (SLL) catch and effort logbooks from foreign licensed tuna fishing vessels in the 1980's. Science staff designed the logbooks and the data were stored in computer flat files accessed by a customized suite of Pascal programms. When MFish Information & Technology Division (ITD) began, official records were archived and **tuna** became the only active source of historical New Zealand SLL tuna catch and effort records which are retained for scientific purposes. Scientific stock management requires the longest available time series to give more weight to Virtual Population Analyses (VPA), reducing probability bandwidths. When ITD began, a new series of data collection forms was instituted. These are now the current TLCER and CELR formats. TLCER data continued to be added to the **tuna** flat files, however some catch and effort data recorded by domestic fishers choosing to use CELR forms were omitted. In 1993 the current design and structure of **tuna** was implemented in an Empress Relational Database Management System (RDBMS). Flat file data were decoded and imported to the more efficient relational structures described in this document. This process has enabled the ability to marry data from TLCER and CELR sources. As other aspects of New Zealand's trilateral commitments to the CCSBT have increased, so has the capacity of **tuna** to include data from other forms including the supplementary RTMP data stored in the table *t_supp*, and the Japanese Yellow Book form in table *t_yellow*.

The data model for **tuna** is an accurate reflection of the real world of commercial tuna fishing within the New Zealand Exclusive Economic Zone (EEZ). Vessels make trips (*t_trip*) within the EEZ under licensing agreements for commercial tuna fishing. While fishing, all vessels are required by law to declare their catch and effort on TLCER or CELR forms. The TLCER form records surface longlining (SLL) effort (*t_tlset*), total catch (*t_tlcatch*) and individual Southern Bluefin Tuna (SBT) processing weights (*t_indcatch*). A table in parallel to *t_tlset*, *t_effort*, holds CELR effort data. Table *t_catch* holds estimated catch data for all methods. Tuna fishing methods mainly comprise SLL, Handlining (HL), Trolling (T), Trot lines or Dahn lines (TL), and Pole & Line (PL). An extra table *t_landing*, holds processing and catch landing records from the MFish Catch Landing Record form (CLR) in the bottom half of the CELR form. This table is used to certify the catch reporting methods of the CELR form where fishers exercise the right to report number or weight of fish caught.

3.2 Database description

The ERD for **tuna** (Figure 1) shows the logical structure of the database and its entities (each entity is implemented as a database *table*) and relationships between these. All of the table's attributes are shown in the ERD. The underlined attributes represent the table's primary key[1]. This schema is valid regardless of the database system chosen, and it can remain correct even if the Database Management System (DBMS) is changed.

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

Each table represents an object, event, or concept in the real world that has been selected to be 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. This schema is valid regardless of the database system chosen, and it can remain correct even if the Database Management System (DBMS) is changed. Most of the tables in the **tuna** database also contain special attributes, called foreign keys[2].

Section 4 shows a listing of all the **tuna** tables as implemented by the Empress RDBMS. 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:

key_name	BTREE	Primary key

This prevents records with duplicate key values from being inserted into the table, e.g., a trip with an existing trip key.

The **tuna** 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 **tuna** are of the type ***one-to-many***[3]. This is shown in the ERD by connecting a single line from the parent table; e.g., *t_trip*, to the child table; e.g., *t_effort*, with crows foot (indicating 'many) pointing to the child. For example, consider the relationship between the tables' *t_trip* (the parent table) and *t_effort* (the child table). Any one trip in *t_trip* can have one or more units of effort in *t_effort*, but any one unit of effort can only be a part of one 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_effort*, which has a mandatory 'one' and an optional 'many'. This means that one trip record can have zero or more (many) units of effort within it, but one unit of effort must have one, and only one, associated record in the trip table.

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.

[2] 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.
[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 survey in *t_survey* can have many strata in *t_stratum* but any one stratum can only come from one survey.
[4] Also known as integrity checks.

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

trip_key	BTREE	Foreign key, refer t_trip(trip_key)

This means that the value of the attribute *trip_key* in the current record 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 example:

vessel_key	BTREE	Normal

Details for each trip are held in the table *t_trip*. Each trip is uniquely identified by a trip code, stored as the attribute *trip_key*.

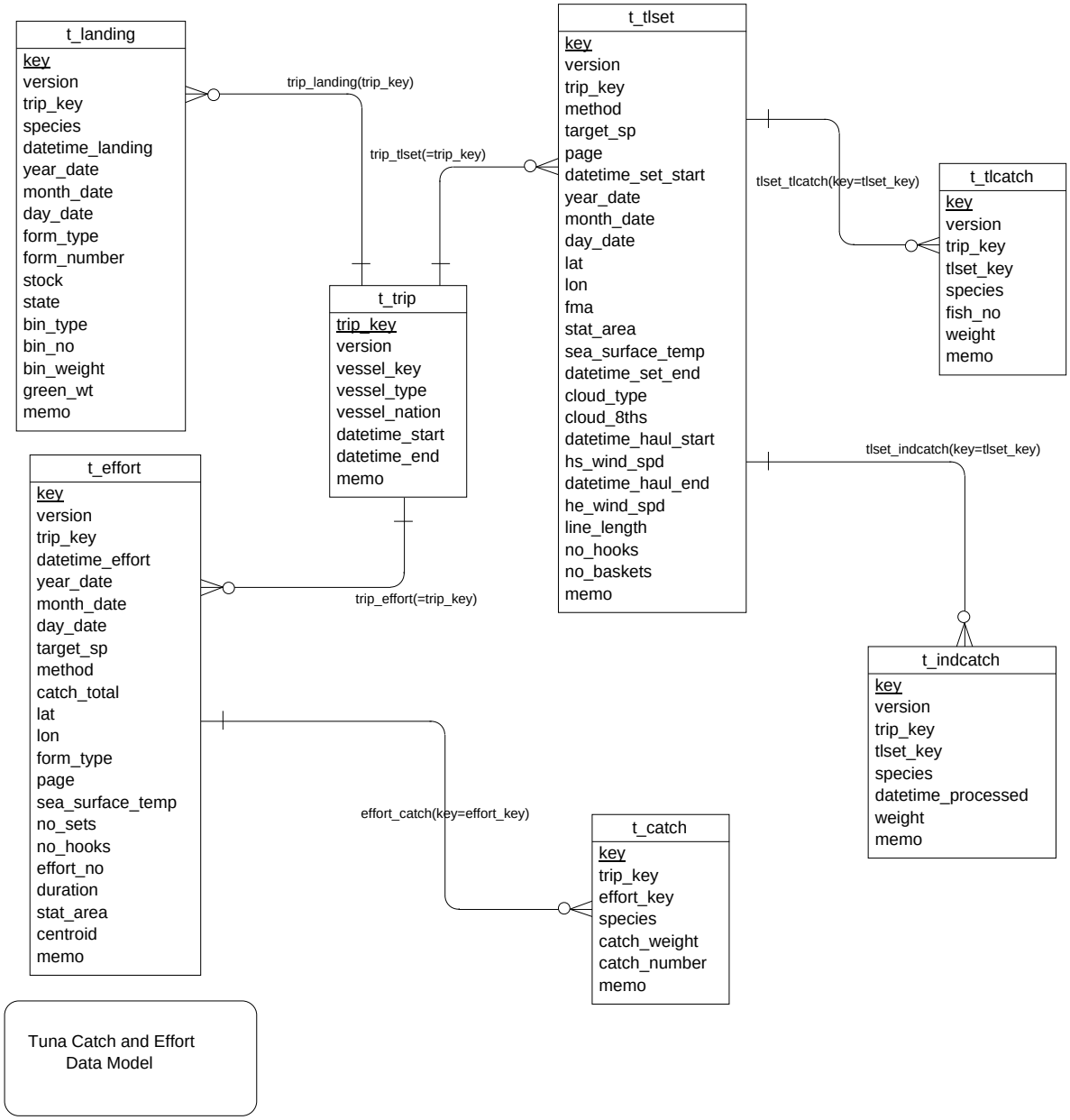
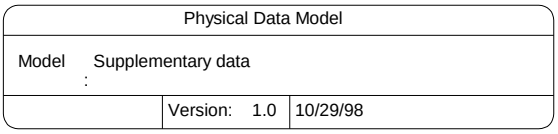


Figure 1: Entity Relationship Diagram (ERD) of the tuna database.

Supplementary data collected from fishers in the RTMP programme, are stored in *t_supp* (Figure 2). *t_supp* holds the length and weight of individual processed SBT and Bigeye tuna (BIG).



t_supp	
callsign	character(7,1)
yr	integer
date	date(10)
activity	smallint
releases	integer
spp	character(3,1)
fish_no	integer
mm	smallint
lgth	integer
w_cond	smallint
wgt	integer
sex	character(1,1)
remarks	character(80,1)

Figure 2: ERD for supplementary data within the tuna database.

Japanese data have been collected and collated into a year by month by 5x5 degree latitude and longitude summary format (Figure 3). These data are made available by the Japanese in exchange for New Zealand data under the terms of the CCSBT agreements.

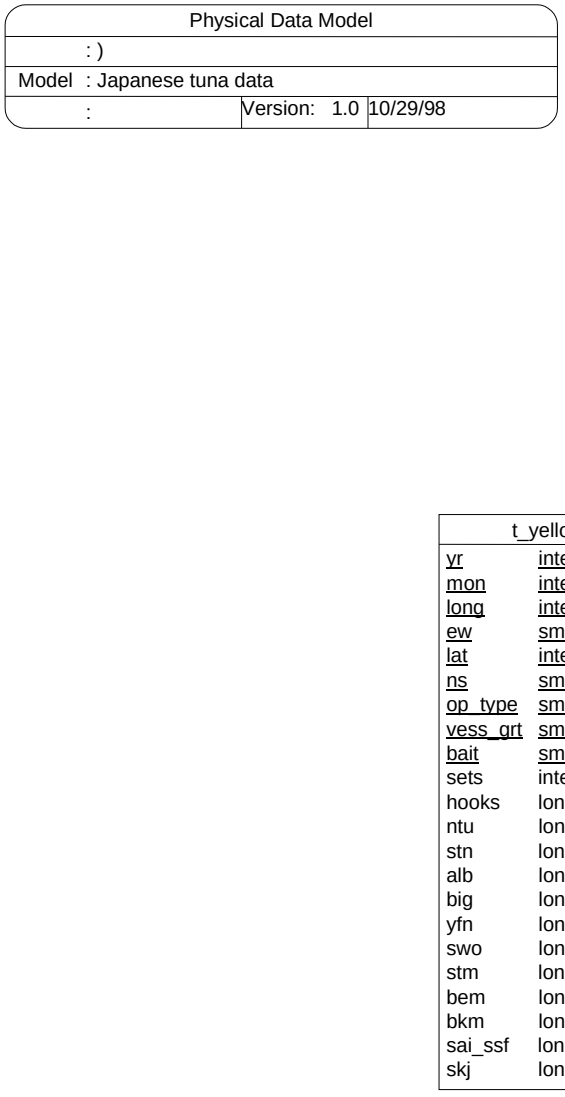


Figure 3: ERD for Japan Yellow Book data within the tuna database

4 Table summaries

The **tuna** database has seven main tables and two supplementary tables.

The following is a listing and brief outline of the main tables contained **tuna**:

- **t_trip :** contains profile information on all trips.
- **t_landing :** contains data filled in a landing return form(CEL or CLR) for each species.
- **t_effort :** contains details of a catch effort filled in a landing return form(CEL or CLR) for each method used.
- **t_catch :** contains catch data per effort for each species.
- **t_tlset :** contains details of each setting of long line filled in TLCER form.
- **t_tlcatch :** contains processed weight and number for each subcatch of each tuna species caught in each set.
- **t_indcatch :** contains processed weight of each individual Southern Bluefin Tuna.

Two supplementary tables inherited from historic tuna database are:

- **t_supp:** contains data collected from fishers in the RTMP programme.
- **t_yellow:** contains Japanese data in exchange for New Zealand under the terms of the CCSBT agreements.

5 Tuna tables

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

5.1 t_trip

Contains header information for each trip event associated with tuna fishing.

Attribute Name	**Attribute Data Type**	**Description**
trip_key	longinteger	Primary key using original trip key value from MFish database
version	smallint	Version of the data in MFish database
vessel_key	longinteger	MFish identification number of the fishing vessel
vessel_type	character(1,1)	Vessel registration type, (**D**omestic, **C**harter, **F**oreign licensed or **U**nknown)
vessel_nation	Character(3,1)	Vessel nationality
datetime_start	time(5)	Start date and time for a trip
datetime_end	time(5)	End date and time for a trip
memo	character(40,1)	Comments added in loading process

Indexed Attribute	**Index type**	**Description**
trip_key	BTREE	Primary key
vessel_key	BTREE	Normal

5.2 t_landing

Contains information for each landing event associated with tuna fishing trip.

Attribute Name	Attribute Data Type	Description
key	longinteger	Primary key generated from a counter
version	smallint	Version of the data in MFish database
trip_key	longinteger	Foreign key to reference *t_trip*
species	character(3,1)	3 character species code
datetime_landing	time(5)	Date and time for the landing event
year_date	integer	Year of landing, convenient for report
month_date	smallint	Month of landing, convenient for report
day_date	smallint	Day of landing, convenient for report
form_type	character(3,1)	3 character code type of the form from which landing data is recorded
form_number	longinteger	Number of the form from which landing data recorded
stock	character(5,1)	Fish stock code
state	character(5,1)	Processed state of fish
bin_type	character(3,1)	Type of container
bin_no	integer	Number of containers
bin_weight	decimal(8,2)	Average weight of each container
green_wt	decimal(8,2)	Green weight of fish
memo	character(40,1)	Comments added in loading process

Indexed Attribute	Index type	Description
key	TIMESERIES	Primary key
trip_key	BTREE	Foreign key, refer *t_trip(trip_key)*

5.3 t_effort

Contains each catch effort in CELR or CLR form associated with tuna fishing.

Attribute Name	Attribute Data Type	Description
key	longinteger	Primary key using event_key value of a fishing event from MFish database
version	smallint	Version of the data in MFish database
trip_key	longinteger	Foreign key to reference *t_trip*
datetime_effort	time(5)	Date and time for a catch effort event
year_date	integer	Year of landing, convenient for report
month_date	smallint	Month of landing, convenient for report
day_date	smallint	Day of landing, convenient for report
target_sp	character(3,1)	Target species
method	character(3,1)	Code for fishing method used.
catch_total	decimal(8,2)	The total weight of Catch for this fishing effort event as estimated at the time.
lat	decimal(6,2)	Decimalised latitude of start of fishing truncated to 1/10th of a degree.
lon	decimal(7,2)	Decimalised longitude of start of fishing truncated to 1/10th of a degree.
form_type	character(3,1)	3 character code type of the form from which landing data is recorded
page	longinteger	Form number from which effort data recorded
sea_surface_temp	decimal(4,1)	Sea surface temperature
no_sets	Integer	Number of sets in the day.
no_hooks	integer	Total hook numbers, may vary depending on method used.
effort_no	integer	Number of catch efforts
duration	decimal(7,4)	Time spent on a catch effort
stat_area	character(5,1)	Statistical area in which the catch effort event started.
centroid	character(1,1)	A calculated area reference point for records that have null lat and lon
memo	character(40,1)	Comments added in loading process

Indexed Attribute	Index type	Description
key	TIMESERIES	Primary key
trip_key	BTREE	Foreign key, refer *t_trip(trip_key)*

5.4 t_catch

Contains each estimated catch record in CELR or CLR form associated with tuna fishing.

Attribute Name	**Attribute Data Type**	**Description**
key	longinteger	Primary key generated from a counter
version	smallint	Version of the data in MFish database
trip_key	longinteger	Redundant key for reference *t_trip*, no foreign key constraint is applied.
effort_key	longinteger	Foreign key to reference *t_effort*.
species	character(3,1)	3 character species code
catch_weight	decimal(10,2)	The weight of catch for this fishing event as estimated at the time.
catch_number	decimal(10,2)	The number of catch for this fishing event as estimated at the time.
memo	character(40,1)	Comments added in loading process

Indexed Attribute	**Index type**	**Description**
key	TIMESERIES	Primary key
effort_key	BTREE	Foreign key, refer *t_effort(key)*

5.5 t_tlset

Contains each tuna catch effort in TLCER form.

Attribute Name	Attribute Data Type	Description
key	longinteger	Primary key using dcf_key value of a fishing event from MFish database
version	smallint	Version of the data in MFish database
trip_key	longinteger	Foreign key to reference *t_trip*
fleet	character(1,1)	Vessel registration type
method	character(3,1)	Code for fishing method used.
target_sp	character(3,1)	Target species
page	longinteger	Form number from which effort data is recorded
datetime_set_start	time(5)	Date and time setting of longline started.
year_date	integer	Year of landing, convenient for report
month_date	smallint	Month of landing, convenient for report
day_date	smallint	Day of landing, convenient for report
lat	decimal(6,2)	Decimalised latitude of start of fishing truncated to 1/10th of a degree.
lon	decimal(7,2)	Decimalised longitude of start of fishing truncated to 1/10th of a degree.
fma	character(3,1)	The fisheries management area in which the fishing event started.
stat_area	character(5,1)	Statistical area in which the catch effort event started.
sea_surface_temp	decimal(4,1)	Sea surface temperature
datetime_set_end	time(5)	Date and time setting of longline finished.
cloud_type	character(5,1)	2 letter code for type of cloud
cloud_8ths	character(5,1)	Numeric code showing number of eighths cloud cover
datetime_haul_start	time(5)	Date and time hauling of longline started
hs_wind_spd	decimal(4,2)	Wind speed(m/s) at time hauling of longline started
datetime_haul_end	time(5)	Date and time hauling of longline finished
he_wind_spd	decimal(4,2)	Wind speed(m/s) at time hauling of longline finished
line_length	decimal(8,2)	Length of long line
no_baskets	integer	Number of baskets
memo	character(40,1)	Comments added in loading process

Indexed Attribute	Index type	Description
key	TIMESERIES	Primary key
trip_key	BTREE	Foreign key, refer *t_trip(trip_key)*

5.6 t_tlcatch

Contains catch data recorded in TLCER form.

Attribute Name	Attribute Data Type	Description
key	longinteger	Primary key generated from a counter
version	smallint	Version of the data in MFish database
trip_key	longinteger	Redundant key for reference *t_trip*, no foreign key constraint is applied.
tlset_key	longinteger	Foreign key to reference *t_tlset*.
species	character(3,1)	3 character species code
fish_no	integer	Number of fish
weight	decimal(16,2)	Processed weight of the fish
memo	character(40,1)	Comments added in loading process

Indexed Attribute	Index type	Description
key	TIMESERIES	Primary key
tlset_key	BTREE	Foreign key, refer *t_tlset(key)*

5.7 t_indcatch

Contains Southern Bluefin Tuna individual catch data recorded in TLCER form.

Attribute Name	Attribute Data Type	Description
key	longinteger	Primary key generated from a counter
version	smallint	Version of the data in MFish database
trip_key	longinteger	Redundant key for reference *t_trip*, no foreign key constraint is applied.
tlset_key	longinteger	Foreign key to reference *t_tlset.*
species	character(3,1)	3 character species code
datetime_processed	time(5)	The start date and time for processing
weight	decimal(16,2)	Processed weight of each Southern Bluefin Tuna.
memo	character(40,1)	Comments added in loading process

Indexed Attribute	Index type	Description
key	TIMESERIES	Primary key
tlset_key	BTREE	Foreign key, refer *t_tlset(key)*

5.8 t_supp

Contains data collected from fishers in the RTMP programme.

Attribute Name	Attribute Data Type	Description
key	longinteger	Primary key generated from a counter
callsign	Character(7,1)	Unique fisher identifier. Radio callsign.
yr	integer	Trip date year.
date	date(10)	Trip date.
activity	smallint	1=fishing, 2=steaming, 3=weather, 4=other.
releases	integer	Number of STN released.
spp	Character(3,1)	Species code.
fish_no	integer	Order lengths reported.
mm	smallint	Length measurement method.
lgth	integer	Length (cm).

w_cond	smallint	Process condition of weighed fish
wgt	integer	Fish weight units=kilograms
sex	Character(1,1)	1=male 2=female 3=unknown 4=not examined
remarks	character(80,1)	Comment field

Indexed Attribute	**Index type**	**Description**
key	TIMESERIES	Primary key
callsign	BTREE	

5.9 t_yellow

Contains Japanese exchange data under the terms of the CCSBT agreements.

Attribute Name	**Attribute Data Type**	**Description**
key	longinteger	Primary key generated from a counter
yr	integer	year
mon	integer	Month
ew	smallint	Longitude code 1=East 2=West
lat	integer	Latitude of Northern (southern) border of the 5 degree grid in northern (southern) hemisphere
ns	Smallint	Latitude code 1=North 2=South
op_type	smallint	Type of operation, 1=SLL
vess_grt	smallint	Vessel size code, 1=20-49 gross tonnes, 2=50-99, 3=100-199, 4=200 plus
bait	smallint	Bait code, 1=Mackerel, 2=squid, 3=others & mixed
sets	integer	Number of sets
hooks	longinteger	Number of hooks
ntu	longinteger	Number of bluefin caught
stn	longinteger	Number of southern bluefin caught
alb	longinteger	Number of albacore caught
big	longinteger	Number of bigeye caught
yfn	longinteger	Number of yellowfin caught
swo	longinteger	Number of broadbill swordfish caught
stm	longinteger	Number of striped marlin caught
bem	longinteger	Number of blue marlin caught
bkm	longinteger	Number of black marlin caught
sai_ssf	longinteger	Number of sailfish & shortbill spearfish caught
skj	longinteger	Number of skipjack caught

Indexed Attribute	**Index type**	**Description**
key	TIMESERIES	Primary key

6 Data Loading and Validation

A loading and validating shell script is used to load and validate tuna fishing data extracted from the MFish database. Because the amount of data involved, each run of this script can only efficiently process a relatively small portion of the data by giving a date window (e.g. several months or half a year).

6.1 Data Loading

- Trip data:
  Data source: X_TRIP_EVENT view from MFish database.
  Select criteria: start trip date time is between specified date time period AND
  form type is one of (CELR, CLR or TLCER) AND
  interpreted version.
- Landing data:
  Data source: X_LANDING view from MFish database.
  Select criteria: start trip date time is between specified date time period AND
  form type is one of (CELR, CLR) AND
  species code is one of TUNA species AND
  interpreted version.
- Effort data
  Data source: X_FISHING_EVENT view from MFish database.
  Select criteria: start effort date time is between specified date time period AND
  form type is one of (CELR, CLR) AND
  interpreted version AND
  (primary fishing method is one of (Trolling, Trot Lines, Pole & Line, Handlining, Surface Longlining, Purse Seining) OR
  Target species is one of TUNA species OR
  (Primary fishing method is not one of (Bottom Longlining, Drop/Dahn Lines)
  AND hook number is not null)
)

- Catch data:
  Data source: X_ESTIMATED_SUBCATCH view from MFish database.
  Select criteria: start fishing date and time is between specified date time period AND
  form type is one of (CELR, CLR) AND
  interpreted version AND
  (primary fishing method is one of (Trolling, Trot Lines, Pole & Line, Handlining, Surface Longlining, Purse Seining) OR
  Target species is one of TUNA species
)

- Tuna Longline Effort data:
  Data source: X_FISHING_EVENT, X_ENVIRONMENT_DATA (for surface temperature data) and X_EVENT (for fisheries management area code) views from MFish database.
  Select criteria: start effort date time is between specified date time period AND
  form type is TLCER AND
  interpreted version.
- Tuna Longline Catch data:
  Data source: X_PROCESSED_CATCH view from MFish Database.
  Select criteria: start processed date time is between specified date time period AND
  form type is TLCER AND
  interpreted version AND
  species code is one of TUNA species.

- Tuna Longline Individual Catch data:
  Data source: X_TUNA_INDIVIDUAL_CATCH view from MFish Database.
  Select criteria: start processed date time is between specified date time period AND
  form type is TLCER AND
  interpreted version AND
  single fish count event.

6.2 Data Integrity Validation

- Trip Record:
  - Original trip code from the MFish database is used as primary key for trip table.
  - Often there are redundant records extracted (multi versions, multi form pages etc.) that need to be found and removed. One and only one trip record should exist in the main trip table for each fishing trip (per vessel and trip date).
  - If a trip record has a null trip code, it will be discarded.
  - If a trip record has no record in the landing or catch effort tables, then it is regarded as not associated with a tuna fishing trip and will be discarded.
- Landing Record:
  - Original event_key from the MFish database is used as primary key for landing table.
  - Keep the latest version for the multi versions of landing record.
  - If a landing record has no trip key value to associate it to a trip, it will be regarded as orphan record, and removed to a temporary table. A restoration is attempted to find the missing trip (the same vessel and trip date), if successful then the newly found trip key is assigned to the orphan record and re-inserted back into landing table.
- Effort Record
  - Original event_key from the MFish database is used as primary key for effort table.
  - Keep the latest version for the multi versions of effort record.
  - If an effort record has no trip key value to associate it to a trip, it will be regarded as orphan record, and removed to a temporary table. A restoration is attempted to find the missing trip (the same vessel and trip date), if successful then the newly found trip key is assigned to the orphan record and re-inserted back into effort table.
- Catch Record
  - An artificial counter is used to generate primary keys for the catch table.

    - Keep the latest version for the multi versions of catch record.
    - If a catch record has no trip key to associate it to a trip, or it has no event key to associate it to an effort record in the effort table, it will be regarded as orphan record, and removed to a temporary table. A restoration is attempted to find the missing trip or event (the same vessel and fishing date time); if successful then the newly found trip key or/and event key is assigned to the orphan record and re-inserted back into catch table.
- Tuna Longline Effort Record
    - Original dcf_key (artificial key corresponding to a form page number) from MFish database is used as primary key for tuna longline effort table.
    - Keep the latest version for the multi versions of tuna longline effort record.
    - If a tuna longline effort record has no trip key value to associate it to a trip, it will be regarded as orphan record, and removed to a temporary table. A restoration is attempted to find the missing trip (the same vessel and trip date), if successful then the newly found trip key is assigned to the orphan record and re-inserted back into tuna longline table. Because there are many tuna longline records from some foreign licensed vessels with no trip information at all, a pseudo trip record is created for each vessel to cope with this special scenario. These pseudo trips have trip key value above 2,000,000,000
- Tuna Longline Catch Record
    - An artificial counter is used to generate primary key for the tuna longline catch table.
    - Keep the latest version for the multi versions of tuna longline catch record.
    - If a tuna longline catch record has no trip key to associate it to a trip, or it has no dcf key to associate it to a tuna longline effort record in tuna longline effort table, it will be regarded as orphan record, and removed to a temporary table. A restoration is attempted to find the missing trip or dcf key (the same vessel and fishing date time), if successful then the newly found trip key or/and dcf key is assigned to the orphan record and re-inserted back into the tuna longline catch table.
- Tuna Longline Individual Catch Record
    - An artificial counter is used to generate primary keys for tuna longline catch table.
    - Keep the latest version for the multi versions of tuna longlineI catch record.
    - If a tuna longline catch record has no trip key to associate it to a trip, or it has no dcf key to associate it to a tuna longline effort record in tuna longline effort table, it will be regarded as orphan record, and removed to a temporary table. A restoration is attempted to find the missing trip or dcf key (the same vessel and fishing date time), if successful then the newly found trip key or/and dcf key is assigned to the orphan record and re-inserted back into tuna longline catch table.

6.3 Business Rule Validation

- Check species codes and reassign if necessary; e.g., BSH reassigned to SHA.
- Check ranges of effort and catch data by method, remove out of range data, the outcome of each business rule validated is recorded in *t_loading* table in **tuna**.
  Data range validation is shown in the following table.

Business Rule No.	Fishing Method	Species	Form Type	Constraint Applied
BR1	Purse seine (PS)	Skipjack tuna (SKJ)	CELR	Catch per set ≤ 160t
BR2	Purse seine trolling	ALB	CELR	Sets per day ≤ 7
BR3		ALB		Number per day ≤ 300
BR4		ALB		Number per trip ≤ 2000
BR5		ALB		Weight of catch per trip ≤ 10t
BR6		ALB		2.0 kg ≤ average weight in landing ≤ 20 kg
BR7		SKJ		1.2 kg ≤ average weight in landing ≤16 kg
BR8		STN		19 kg ≤ average weight in landing ≤ 225kg
BR9		NTU, TOR		19 kg ≤ average weight in landing ≤ 350 kg
BR10		YFN		1.2 kg ≤ average weight in landing ≤ 70 kg
BR11	longline	ALB		2 kg ≤ average weight in landing ≤ 45 kg
BR12	longline	BIG		19kg ≤ average weight in landing ≤ 210 kg
BR13	longline	NTU, TOR		19 kg ≤ average weight in landing ≤ 350 kg
BR14	longline	YFN		14 kg ≤ average weight in landing ≤ 200 kg
BR15		STN		15 kg ≤ average weight in landing ≤ 250 kg
BR16		SWO		16 kg ≤ average weight in landing ≤ 650 kg
BR17		ALL		sets per day ≤ 2
BR18		ALL		50 ≤ hooks per set ≤ 4000

6.4 Data Range Validation

- Nullify temperature field where the temperature is not between 10 °C ~ 30 °C.
- Update records with null latitude or/and longitude to stat area centroid.
- Nullify number of hooks field for those records where method is trolling, pole & line, or handlining, and number of hooks is less than 25.

7 Acknowledgements

The author would like to thank Talbot E. Murray for his making the business rules applied on tuna data,

and Dave Banks for review and editorial comment, and Kevin MacKay for his technical input for this document. Part of the document is based on the H.A. Dean's original tuna database document.

8 References

Dean, H.A. 1998: Database documentation: tuna. *NIWA Internal Report No. 37*. 27p.

Ng, S. 1992: Standards for setting up databases and their applications. *MAF Fisheries Greta Point Internal Report No. 180*. 31p.

.