Database documentation: vessel

D. O. Fisher, C. Sutton, and F. Wei

NIWA Fisheries Data Management
Database Documentation Series

Updated Jan 2016

Revision History

Version	Change	Date	Responsible
0.1	Initial version	10 Jan 2005	David Fisher
0.2	Incorporating comments from G Straker, MFish 21Mar2005	March 2005	David Fisher
1.0	First official release	17 May 2005	David Fisher
1.1	Updates to Appendix 2 Rules and expanding explanation of grooming process.	26 Sep 05	Colin Sutton & David Fisher
1.2	Updates based on comments from M Vignaux, MFish.	28Oct05	David Fisher
2.0	Postgres version	Jan 2016	D Fisher, F Wei

Contents

List of Figures

1 Introduction to the 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 for Primary Industries (MPI) formerly the Ministry of Fisheries.

This MPI data set, incorporates historic research data, data collected by MAF Fisheries prior to the split in 1995 of Policy to the Ministry of Fisheries and research to NIWA, and data collected by NIWA and other agencies for the Ministry of Fisheries and subsequently for MPI.

This document describes the vessels specifications database **vessel**, and is 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 link together.

This document is intended as a guide for users and administrators of the **vessel** database.
This database has been implemented as a schema within the Postgres database called **fish**.

Access to this database is restricted to specific nominated personnel as specified in the current Data Management contract between the Ministry for Primary Industries and NIWA. Any requests for data should in the first instance be directed to the Ministry.

2 Vessel Registrations

Since the 1970's it has been mandatory for vessels fishing in New Zealand waters to be registered. Registration includes the completion of a form detailing the specifications of the vessel, including the size of the vessel, year built, and engine(s) make and model. The Ministry of Fisheries and its predecessors have held these registration data.

Fishing vessels may change their name or call sign, or re-power with a new engine with a different kilowatt (or horsepower) rating. Vessels may come and go from NZ waters. In the early years of New Zealand's fisheries foreign licensed vessels came to NZ waters to fish within the EEZ. Keeping track of individual vessel hulls, particularly if they return to New Zealand under a different name can be a difficult process and errors have been made in the past in assigning various vessel identifiers to generate a unique value for each individual vessel hull.

The objective of this database is to create a research version of these vessel registration data, for use by fisheries scientists in their analysis of catch and effort data from various New Zealand fisheries. It is also likely that these data will be useful to fisheries managers and compliance staff from the Ministry of Fisheries. This database provides the best-groomed dataset of individual fishing vessels operating in New Zealand waters to date.

2.1 Data sources

MFish provided NIWA with 3 overlapping vessel datasets.

1. An initial excel spreadsheet that included Lloyds IMO numbers that had been added by a student employed by MFish.
2. Data from the MFish vessel database based on the 'old form' used for vessel registrations.
3. Data from the MFish vessel database based on the 'new form' used for vessel registrations.

No one dataset contained all attributes or the full time period (i.e., all datasets contained one or more all null columns and/or did not cover the complete time period). The initial datasets included approximately one record per vessel per year per dataset. The scope of the contract was to groom data for vessels over 20 metres overall length and registered between October 1983 and September 2003.

2.2 Data validation

These three datasets referred to above were 'collapsed' separately. That is the records were sorted by the MFish vessel key and then by spec_from and spec_to attributes. These spec_from and spec_to attributes contain the start date and end date applicable to each record. Each consecutive record was compared for each vessel and if all attributes were identical except spec_from and spec_to then the minimum spec_from and maximum spec_to dates were assigned to the 'collapsed' or combined record. If any attribute changed then a record with these specification dates was written out and the process continued. This process was repeated for all records for each vessel.

We then loaded these three datasets to the original vessel table, *t_vessel_org* in the database. These data were extensively groomed based on the rules in the *t_rule* table and in Appendix 2. Initially, this grooming process was to identify individual vessel hulls. We used the four main identifying attributes: MFish vessel_key, vessel_id, call_sign and vessel_name, plus combinations of other attributes to determine those records that represented the same vessel. A new attribute vid (NIWA vessel ID) was created to identify individual vessels.

A series of electronic routines and manual checking of marginal cases was undertaken to assign distinct vid values to individual vessels.

The Lloyds Register of Shipping database on CD-ROM from 2004 was searched by vessel name and call sign for matching vessels where the imo_no was missing (or null) in our dataset. Where an imo_no was found for a matching vessel this was updated in our dataset. Most of the records with a missing imo_no in the final dataset were less than 100 t gross tonnage, and so were not included in the Lloyds database. Cross checks were made looking for inconsistencies such as where 2 records with the same vid had a different IMO number or 2 records had the same IMO number but different vid. These cases were manually checked and appropriate corrections made. This process of assigning Lloyds numbers and electronically and manually cross checking records, in some cases, resulted in merging what were recorded as two hulls into one hull.

Where two or more MFish vessel_key values had been assigned to the same individual vessel, identified by a distinct vid, the vessel_key that was the smallest number was adopted. This should equate to the vessel_key that was assigned first.

The data grooming process followed the rules in Appendix 2. For each vessel identified by a vid, records were compared electronically and where there were different values for the same attribute these rules were applied. Null values were populated based on not null values in other records for the same attribute for the same vid, as per the rules in Appendix 2.

MFish had recorded the spec_to as 31 Dec 2999 (or 2099) when the vessel was not known to have left NZ waters for that registration record. If a record had a spec_to of 31 Dec 2999 in the middle of a consecutive sequence for one dataset (i.e. one owner_key) then the spec_to was set to equal the next spec_from value.

In creating this research version of these vessel registration data, we did not retain information such as when a vessel was not registered to fish in NZ waters. This decision is based on the assumption that fisheries research scientists want to know the vessel details when it was fishing NZ waters and are not concerned if it was not fishing in NZ waters.

As a general rule, decimalised values for overall_ length, breadth, draught, gross_tonnage and kilowatts were rounded down to the nearest whole integer.
The data were 'collapsed' again at the end of this electronic grooming process.

Final manual grooming was then undertaken, which included comparing remaining records for each vid. For those records in the *t_vessel* table where an imo_no value existed these records were compared with the data on the Lloyds Register of Shipping CD-ROM. The Lloyds database contains the full history of each vessel including previous names and changes of engine. The details of this final manual grooming are included in Appendix 2 – Rules.

Approximately 10%, i.e., 150 vessels were checked visually against the *t_vessel_org* and *t_history* tables. This was to ensure that the grooming process had worked appropriately. No obvious discrepancies were apparent.

Any remaining null values were then populated electronically from the Lloyds data in the table t_lloyds where appropriate.

A final 'collapse' was undertaken so in some cases the final groomed dataset in the t_vessel table contains one record for each vessel hull. Where genuinely different values exist for one vessel hull at different times, such as the vessel was re-engined with a more powerful engine, then 2 or more records will exist for that vessel hull.

Where there is more than one record for a vid or vessel hull in *t_vessel*, ideally the dataset should contain only one record that relates to any given date, defined by a pair of spec_from and spec_to dates. Users of these data should be aware that, given the variable data quality of the source data and that the final dataset results from merging three source datasets, in some cases selecting a vessel on a date results in more than one record in *t_vessel*. In these cases we recommend that researchers examine the 'duplicate' results and in some cases the attributes they are interested in may be the same so a unique data subset may be chosen to satisfy their requirements. If the attributes of interest vary then some other method such as choosing the earliest or minimum/maximum values may be appropriate.

3 Data Structures

3.1 Table relationships

The vessel database contains several tables. The ERD for **vessel** (Figure 1) shows the logical structure (i.e. schema) of the database and its entities (each entity is implemented as a database *table*) and the relationships between these tables and tables in other databases. This schema is valid regardless of the database system chosen, and it can remain correct even if the Database Management System (DBMS) is changed. Each table represents an object, event, or concept in the real world that is 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 attribute represents the table's primary key[1].

Some of the tables in the **vessel** database have attributes called foreign keys[2]. The foreign keys define the relationships between the tables in **vessel**.

The **vessel** database is implemented as a relational database; i.e., 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. There are three types of relationships possible between tables, but only one exists in **vessel**: one-to-many[3]. These relationships can be seen in ERDs by connecting a crows foot (indicating 'many') from the child table; e.g., *t_vessel_org*, to the parent table; e.g., *t_meta,* with a straight line (indicating 'one') pointing to the parent.

Every relationship has either a mandatory or optional aspect to it. If a relationship is mandatory, then it has to occur at least once, while an optional relationship might not occur at all. For example, in Figure 1, consider the relationship between the table *t_meta* and it's child table *t_vessel_org.* The symbol 'o' by the child table *t_vessel_org* means that a metadata record (in table *t_meta*) can have zero or many original vessel records (in table *t_vessel_org)*, while the bar by the parent table *t_meta* means that for every original vessel record there must be a matching record in *t_meta*.

---

[1] A primary key is an attribute or a combination of attributes that contains an unique value to identify that 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 landing in *t_landing* can have many catches in *t_catch* but one catch can only come from one landing.

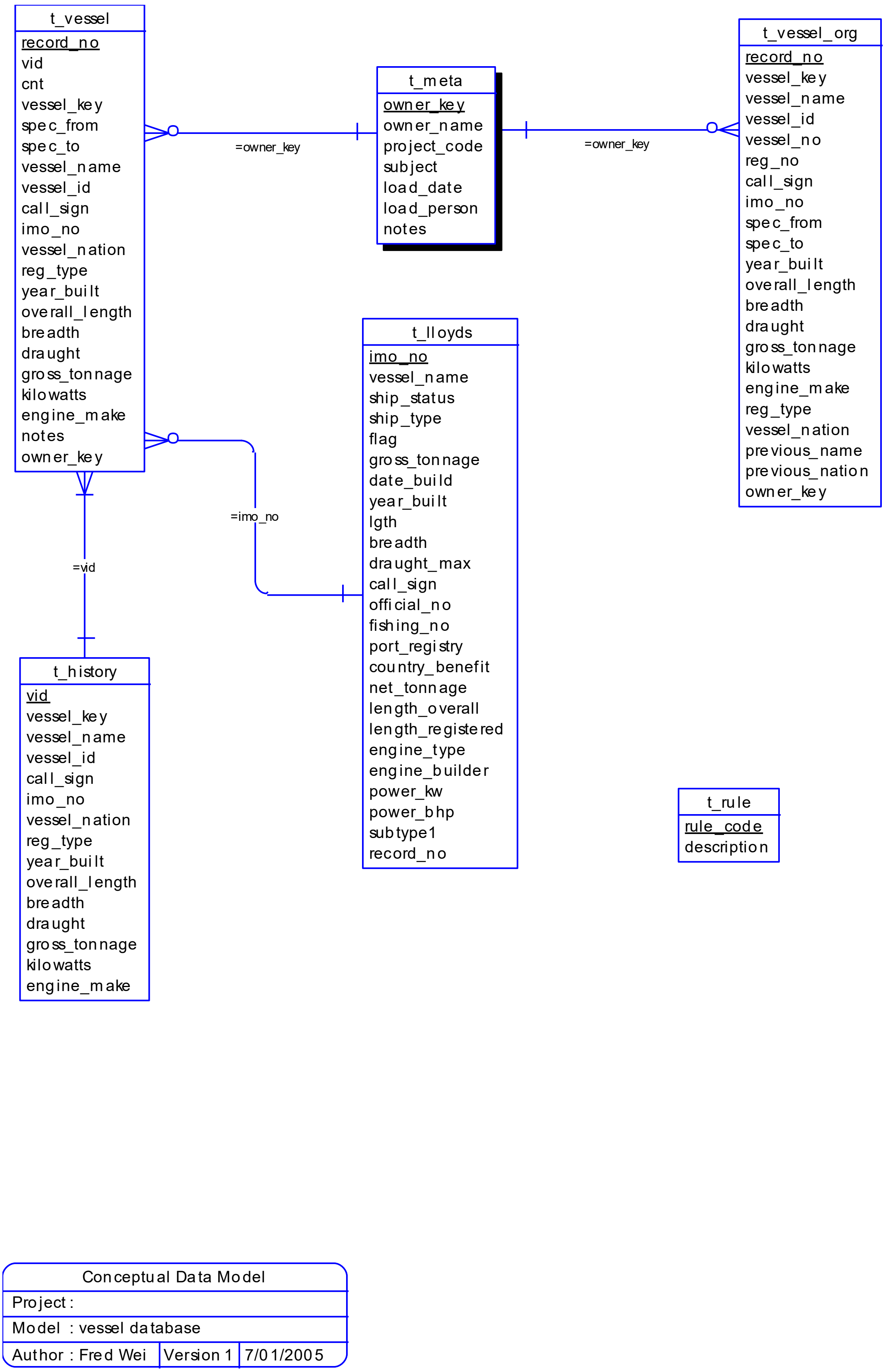

Figure 1: Entity Relationship Diagram (ERD) for the **vessel** database.

These links are enforced by foreign key constraints[4]. These 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 either a parent record is deleted, the parent record is altered so the relationship is lost, or a child record is entered without a parent record

Foreign key constraints are shown in the table listings by the following format:

```
Foreign-key constraints:
    "foreign key name" FOREIGN KEY (attribute[,attribute]) REFERENCES
      parent table (attribute[, attribute])
```

Note that the typographical convention for the above format is that square brackets [] may contain more than one item or none at all. Items stacked between vertical lines | | are options of which one must be chosen.

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

Foreign-key constraints:
"fk_t_vessel_t_meta" FOREIGN KEY (owner_key) REFERENCES t_meta(owner_key)

This means that the value of the attribute *owner_key* in the current record must already exist in the parent table *t_meta* or the record will be rejected and an error message will be displayed:

For tables residing in external databases, the parent table name will be prefixed by the name of the database.

Section 5 lists all the **vessel** tables as implemented by the Postgres DBMS. As can be seen in the listing of the tables, each table has a primary key that forms a unique index on the attribute. Primary keys are generally listed using the following format:

Indices: index_name PRIMARY KEY, btree (*attribute* [, *attributes*])

where attribute(s) make up the primary key and the index name is the primary key name. Primary keys prevent records with duplicate keys from being inserted into the database tables; e.g., a record with vessel_id number that already exists in the table.

The database listing (Tables 1-6) show that the tables also have indices on some attributes. 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: index_name btree (*attribute*)

Note that indices may be simple (pointing to one attribute) or composite (pointing to more than one attribute).

[4] Also known as referential constraints or integrity checks

3.2 Database design

The top-level table in **vessel** is *meta* (Table 1). This table holds some summary information (metadata) for this database. The primary key of this table is owner_key, which identifies the source of the data to one of the three source datasets.

The owner_key provides a link to the *t_vessel_org* table, which contains all the original data from the three MFish datasets.

These data from the *t_vessel_org* table were groomed using a series of electronic and manual processes. The resulting groomed data are contained in the t_vessel table and are considered to be the best version to use for research purposes.

The *t_rule* table documents the rules applied to the data between the *t_vessel_org* and *t_vessel* tables. The rules that were applied to groom data as part of this process are documented as a series of rule codes separated by the '|' character in the notes attribute in the t_vessel table.

The *t_history* table lists all variations recorded for each attribute for each vessel. With only one record for each unique vid, multiple values for each attribute are separated by the '/' character. This table allows users to select, for example, all vessels that have had multiple vessel keys assigned by selecting "where vessel_key like '%/%".

The *t_lloyds* table contains the current data extracted from the Lloyds Register of Shipping database in 2004. These Lloyds data have been used in the grooming process as a reference dataset and can be joined with the *t_vessel* table using the imo_no attribute. The Lloyds dataset only holds information on vessels with a minimum gross tonnage of 100 tonnes.

4 Table Summaries

The **vessel** database has five tables containing vessel specifications data, plus one table which documents the rules used in grooming the vessel specification data and one view.

The following is a listing and brief outline of the tables contained in **vessel**:

1. **t_meta**: contains data ownership information for the database
2. **t_vessel**: contains the groomed details for each vessel.
3. **vessel_org**: contains the 3 original datasets supplied by MFish.
4. **t_history**: includes all values for each attribute for each vessel hull in 1 record.
5. **t_lloyds**: contains vessel details from Lloyds register of shipping.
6. **t_rule**: documents the rules used to groom the vessel data and the associated codes used in the notes attribute of table *t_vessel*.
7 **v_vessel**: View of t_vessel groomed vessel dataset, omitting sensitive vessel id columns.

5 vessel Tables

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

5.1 Table 1: t_meta

Comment: This table contains data summary information for the database.

Column	Type	Null?	Description
owner_key	smallint	No	Number as primary key.
owner_name	character varying(32)		Name of the dataset owner.
project_code	character varying(32)		The project code associated with the dataset.
subject	character varying(64)		Descriptive text relating to the dataset.
load_date	date		Date when the dataset is loaded into the database.
load_person	character varying(32)		Name of the person who loaded the dataset.
notes	character varying(512)		Any commentary text regarding the dataset.

```
Indexes:
   "pk_t_meta" PRIMARY KEY, btree (owner_key)
```

5.2 Table 2: t_vessel

Comment: This table contains the groomed vessel dataset.

Column	Type	Null?	Description
record_no	integer	No	Unique identification number for each record in this table.
vid	integer	No	Unique identification number assigned by NIWA for each vessel.
cnt	smallint		The number of records (count) collapsed into the current record in the grooming process.
vessel_key	integer		MFish assigned number to identify a vessel.
spec_from	date		Starting date for the vessel specifications.
spec_to	date		Finishing date for the vessel specifications.
vessel_name	character varying(32)		Vessel name.
vessel_id	character varying(8)		MFish assigned alphanumeric code to identify a vessel, typically registration number or call sign.
call_sign	character varying(10)		Signal letters or radio call sign assigned by the relevant Registration (Flag) Authority.
imo_no	integer		Unique International Maritime Organisation number assigned by Lloyd's Register to each ship.
vessel_nation	character varying(3)		3 character code for vessel's flag or nationality.
reg_type	character varying(1)		MFish assigned classification code - Charter, Domestic, Foreign, Unknown.
year_built	smallint		The year the vessel was built.
overall_length	smallint		Overall length in metres.
breadth	smallint		Breadth in metres.
draught	smallint		Draught in metres.
gross_tonnage	smallint		Gross tonnage.
kilowatts	smallint		Engine power in kilowatts.

engine_make	character varying(32)	Engine make.
notes	character varying(256)	Rules used in grooming process and other comments separated with '|'. Not arranged in the order they were applied.
owner_key	smallint	Foreign key link to t_meta record.

```
Indexes:
   "pk_t_vessel" PRIMARY KEY, btree (record_no)
   "nx_t_vessel_vessel_key" btree (vessel_key)

Foreign-key constraints:
   "fk_t_vessel_t_history" FOREIGN KEY (vid)
    REFERENCES vessel.t_history(vid)
   "fk_t_vessel_t_meta" FOREIGN KEY (owner_key)
    REFERENCES vessel.t_meta(owner_key)
```

5.3 Table 3: t_vessel_org

Comment: This table contains the original vessel datasets from MFish.

Column	Type	Null?	Description
record_no	integer		Unique identification number.
vessel_key	integer		MFish assigned number to identify a vessel.
vessel_name	character varying(32)		Vessel name.
vessel_id	character varying(8)		MFish assigned alphanumeric code to identify a vessel, typically registration number or call sign.
vessel_no	character varying(8)		MFish assigned number to identify a vessel.
reg_no	character varying(16)		Vessel registration number.
call_sign	character varying(10)		Signal letters or radio call sign assigned by the relevant Registration (Flag) Authority.
imo_no	integer		Unique International Maritime Organisation number assigned by Lloyd's Register to each ship.
spec_from	date		Starting date for the vessel specification.
spec_to	date		Finishing date for the vessel specification.
year_built	smallint		The year a vessel is built.
overall_length	numeric(6,1)		Overall length in metres.
breadth	numeric(4,1)		Breadth in metres.
draught	numeric(4,1)		Draught in metres.
gross_tonnage	numeric(7,1)		Gross tonnage.
kilowatts	numeric(5,1)		Engine power in kilowatts.
engine_make	character varying(32)		Engine make.
reg_type	character varying(1)		MFish assigned classification code - Charter, Domestic, Foreign, Unknown.
vessel_nation	character varying(3)		Vessel flag or nationality.
previous_name	character varying(32)		Previous name of vessel.

previous_nation	character varying(3)	Previous flag or nationality of vessel.
owner_key	smallint	Foreign key link to t_meta record.

```
Foreign-key constraints:
   "fk_t_vessel_org_t_meta" FOREIGN KEY (owner_key)
    REFERENCES vessel.t_meta(owner_key)
```

5.4 Table 4: t_history

Comment: This table lists all values of each attribute separated by '/' for each vessel.

Column	Type	Null?	Description
vid	integer	No	Unique identification number assigned by NIWA for each vessel.
vessel_key	character varying(32)		MFish assigned number to identify a vessel.
vessel_name	character varying(128)		Vessel name.
vessel_id	character varying(128)		MFish assigned alphanumeric code to identify a vessel, typically registration number or call sign.
call_sign	character varying(128)		Signal letters or radio call sign assigned by the relevant Registration (Flag) Authority.
imo_no	character varying(32)		Unique International Maritime Organisation number assigned by Lloyd's Register to each ship.
vessel_nation	character varying(32)		3 character code for vessel flag or nationality.
reg_type	character varying(8)		MFish assigned classification code - Charter, Domestic, Foreign, Unknown.
year_built	character varying(64)		The year a vessel is built.
overall_length	character varying(64)		Overall length in metres.
breadth	character varying(64)		Breadth in metres.
draught	character varying(64)		Draught in metres.
gross_tonnage	character varying(64)		Gross tonnage.
kilowatts	character varying(128)		Engine power in kilowatts.
engine_make	character varying(128)		Engine make.

Indexes:
"pk_t_history" PRIMARY KEY, btree (vid)

5.5 Table 5: t_lloyds

Comment: This table contains selected information from the Lloyd's Register of Shipping record.

Column	Type	Null?	Description
imo_no	integer	No	Unique International Maritime Organisation number assigned by Lloyd's Register to each ship.
vessel_name	character varying(32)		The current name of the ship.
ship_status	character varying(32)		In Service Commission(S), Laid Up(L), In Casualty or Repairing(R), Converting / Rebuilding(C), To Be Broken Up(T), Unconfirmed Ships(X).
ship_type	character varying(16)		Bulk Carrier, Container Ship, Dredger, Fishing, General Cargo Ship, Icebreaker, (Offshore) Supply Ship, Passenger Ship, Refrigerated Cargo Ship, RoRo Cargo Ship, Tanker, Vehicles Carrier.
flag	character varying(32)		Indicates the flag country of registry under which the ship normally operates.
gross_tonnage	smallint		Gross tonnage.
date_build	character varying(16)		The year and month when the new construction survey process is completed.
year_built	smallint		Reported year of completion of construction.
lgth	numeric(6,2)		Overall Length between perpendiculars else Registered length.
breadth	numeric(5,2)		Extreme Breadth else Moulded Breadth of the vessel.
draught_max	numeric(5,2)		In most cases this is the maximum summer draught amidships.
call_sign	character varying(16)		Signal letters or radio call sign assigned by the relevant Registration (Flag) Authority.
official_no	character varying(16)		The identification number assigned by the national authority.

fishing_no	character varying(16)	The identification number assigned by the national authority to ships engaged in the fishing industry.
port_registry	character varying(32)	Place where the ship is registered. Home port is shown where there is no port of registry.
country_benefit	character varying(32)	The country considered to be the main beneficiary from the earnings generated by the operation of the vessel.
net_tonnage	smallint	Represent a measure of the ship's freight earning capacity.
length_overall	numeric(6,2)	The extreme length of the ship.
length_registered	numeric(6,2)	Measured from the extreme fore point of the hull to the after end of the stern post, or if there is no stern post to the fore side of the rudder stock.
engine_type	character varying(32)	Oil, Steam Turbine or Steam Reciprocating.
engine_builder	character varying(128)	Manufacturer of the main engine.
power_kw	smallint	Power in kilowatts. Power for LR Class ships is Design Power; for non-LR Class Service Power is usually recorded.
power_bhp	smallint	Power in bhp. Power for LR Class ships is Design Power; for non-LR Class Service Power is usually recorded.
subtype1	character varying(32)	Subtypes of vessel, e.g. if fishing or reefer, and types of fishing vessel by method.
record_no	integer	Non-lloyds field, unique number assigned while loading the data.

```
Indexes:
   "pk_t_lloyds" PRIMARY KEY, btree (imo_no)
   "nx_t_lloyds_call_sign" btree (call_sign)
   "nx_t_lloyds_vessel_name" btree (vessel_name)
```

5.6 Table 6: t_rule

Comment: This table contains the rules used in the grooming process.

Column	Type	Null?	Description
rule_code	character varying(16)		Alphanumeric Code for a rule.
description	character varying(1024)		Description of the rule.

5.7 View 1: v_vessel

```
Comment: View of t_vessel groomed vessel dataset, omitting sensitive vessel
         id columns.

Column          Type

record_no       integer

cnt             smallint

vessel_key      integer

spec_from       date

spec_to         date

vessel_nation   character varying(3)

reg_type        character varying(1)

year_built      smallint

overall_length  smallint

breadth         smallint

draught         smallint

gross_tonnage   smallint

kilowatts       smallint

engine_make     character varying(32)

notes           character varying(256)

owner_key       smallint

View definition:
SELECT t_vessel.record_no, t_vessel.cnt, t_vessel.vessel_key,
t_vessel.spec_from, t_vessel.spec_to, t_vessel.vessel_nation,
t_vessel.reg_type, t_vessel.year_built, t_vessel.overall_length,
t_vessel.breadth, t_vessel.draught, t_vessel.gross_tonnage,
t_vessel.kilowatts, t_vessel.engine_make, t_vessel.notes,
t_vessel.owner_key
  FROM vessel.t_vessel;
```

6 vessel business rules

6.1 Introduction to business rules

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

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. These rules state that a value **must** meet the specified criteria.

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.

6.2 Summary of rules

Vessel details (t_vessel)

record_no	Must be a unique integer.
vid	Must be an integer greater than zero, and must be a unique value for a specific hull.
cnt	Must be an integer greater than zero.
vessel_key	Must be an integer greater than zero, and must be a unique value for a specific hull.
spec_from	Must be a valid date, and should be between 1978 and the current date.
spec_to	Must be a valid date, and should be between 1978 and the current date, or 2099, or 2999. **Multiple column checks on spec dates:** The spec_from date must not be greater than the spec_to date.
vessel_id	Should consist only of the characters A-Z and 1-9.
call_sign	Should be a valid NZ or international radio call sign. Typically 4 to 7 alpha numeric characters.
imo_no	Must be an integer greater than zero.
vessel_nation	Should be a valid MFish 3 character code for vessel nationality.
reg_type	Must be a 1 character code, and should be one of ('C','D','F','U').
year_built	Must be an integer greater than zero, and should be a valid year between 1900 and the current year.
overall_length	Must be an integer greater than zero, and should be between 20 and 150 metres.
breadth	Must be an integer greater than zero, and should be between 3 and 24 metres.
draught	Must be an integer greater than zero, and should be between 1 and 12 metres.
gross_tonnage	Must be an integer greater than zero, and should be between 1 and 9000.
kilowatts	Must be an integer greater than zero, and should be between 40 and 9000.
notes	Should contain codes separated by '|', as documented in the *t_rule* table

Metadata (t_meta)

owner_key   Must be a unique integer.

load_date   Must be a valid date, and should be between 2004 and the current date.

History data (t_history)

vid   Must be an integer greater than zero, and must be a unique value for a specific hull.

Rules details (t_rule)

rule_code   Rule code must be unique.

For the Lloyds reference data in table *t_lloyds*, as supplied by Lloyds Register of Shipping, business rules are not applied other than enforcing numeric data types. This is because the Lloyds data are a reference data set and were not groomed or checked before loading to this table.

For the original data in table *t_vessel_org*, as supplied by MFish, business rules are not applied other than enforcing numeric and date data types. This is to retain the original data as supplied by MFish.

7 Acknowledgments

The authors would like to thank Peter Shearer for his review and editorial comment for this document.

Appendix 1 Reference code tables

Nationality	**Nationality type description**
AUS	Australia
BZE	Belize
CHI	China (People's Republic of)
COO	Cook Islands
FIJ	Fiji
GRE	Greece
JAP	Japan
KOR	Korea
NOR	Norway
NZL	New Zealand
PHI	Philippines
POL	Poland (Republic of)
RUS	Russian Federation
SNG	Singapore (Republic of)
SVG	Saint Vincent
TAI	Taiwan
UKR	Ukraine
USA	U.S.A
VAN	Vanuatu (Republic of)

Codes used for the reg_type attribute, documenting the registration type.

Vessel reg type	Vessel reg type description
C	Charter
D	Domestic
F	Foreign License
U	Unknown

Appendix 2 Data Grooming Rules

The rule codes used in the notes attribute of the t_vessel table are documented below.
These rules are also documented in the t_rule table.

As a general rule, decimalised values for overall_ length, breadth, draught, gross_tonnage and kilowatts were rounded down to the nearest whole integer.
Rules 202 to 207 are applied by script, and 211 to 250 are used in the manual grooming process.
These numeric rules were applied in sequence. The notes in table t_vessel do not appear in this order because the notes fields for several records were then merged.

rule_code	description
202	If a vessel record has no common id attribute to other records in the merged dataset, it is treated as an individual vessel.
203	Two vessels are the same vessel if both have one common id attribute and either a. the same imo number or b. the same built year and overall length and tonnage and breadth and draught.
204	Two vessels are the same vessel if both have two common id attributes and either a. the same imo number or b. the same year built and overall length and tonnage and breadth and draught.
205	Two vessels are the same vessel if both have three common id attributes and either a. the same imo number or b. the same built year and overall length and tonnage.
206	Two vessels are the same vessel if both have four common id attributes, ie vessel_key, vessel_id, call_sign and vessel_name.
207	Two vessels are the same vessel if both have three common id attributes and one null id attribute and either a. the same imo number or b. two out of the three of the following attributes built year, length, tonnage.
208-211	Rules 208, 209, 210 & 211 do not exist.
212	If 6 attributes agreed then only one hull was considered to exist for the two sets of data.
213	If 5 attributes agreed (provided one of the attributes that did not agree related to vessel_name, call_sign, or a dimension, and was minor) then one hull existed for the two sets of data.
214	If 4 attributes agreed (provided at least 1 of the attributes that did not agree met one of the rules listed in 213) then one hull existed for the two sets of data.
215	If 4 attributes agreed, and it was not possible to compare other attributes (due to the presence of null values) then only one hull existed for the two sets of data, but only if there was compelling data to support this assumption. For example, imo numbers were the same. It was decided that 4 was the minimum number of attributes required for comparison, as any less meant that it was not possible to make a meaningful conclusion.
216	In the instance that the conditions in rule 215 are not met, or fewer than 4 attributes were present it was assumed that 2 hulls existed.
232	Same as 212 except there are three agreed ID attributes (vessel_name, call_sign, vessel_id, vessel_key).
233	Same as 213 except there are three agreed ID attributes (vessel_name, call_sign, vessel_id, vessel_key).
234	Same as 214 except there are three agreed ID attributes (vessel_name,call_sign,vessel_id,vessel_key).
235	Same as 215 except there are three agreed ID attributes (vessel_name,call_sign,vessel_id,vessel_key).
236	Same as 216 except there are three agreed ID attributes

(vessel_name, call_sign, vessel_id, vessel_key).

242	Same as 212 except there are three agreed ID attributes and one null ID attribute.
243	Same as 213 except there are three agreed ID attributes and one null ID attribute.
244	Same as 214 except there are three agreed ID attributes and one null ID attribute.
246	Same as 216 except there are three agreed ID attributes and one null ID attribute.

IMrNEW	added the imo_no based on the Lloyds Register of Shipping database searching Lloyds by vessel name or call sign, and then on other matching attributes, including ship type, overall length, year built etc.

The following rules were applied after the (mostly numeric) rules above, and the order they were applied does not affect the final groomed data.
These following rules were only applied to vessels with the same vid
(i.e. we consider they were the same vessel)

BR	rules applied to breadth
CL	rules applied to collapse procedure
CS	rules applied to call sign
DR	rules applied to draught
EM	rules applied to engine maker
GT	rules applied to gross tonnage
IM	rules applied to imo number
KW	rules applied to kilowatts
NT	rules applied to vessel nationality
OL	rules applied to overall length
RT	rules applied to vessel registration type
VD	rules applied to vessel id of MFish
VK	rules applied to vessel key of MFish
VN	rules applied to vessel name
YB	rules applied to year built

EMrMAN	Manually simplify engine maker.
EMrSTN	for engine make string, first char of each word set upper case.
GTrOv	if gross tonnage > 8000t, set to null.
IMrDIS	same vid got different IMO numbers with 1 digit different, manual fix with IMO number checked against Lloyds and imo_no 7405338 corrected to 7405388.
IMrDUP	different VIDs have the same IMO number, manual fix with vessel attributes checked against Lloyds and vid corrected where appropriate.
IMrNU	use unique IMO no to fill in all the nulls.
RTrU	code in C/D/F/U, set to U for unknown or null if only one code exists except U, then set all to the code.
VKrDUP	different VIDs have the same vessel key, manual fix.
VNrSTN	for name string, a. first char of each word set upper case; b. for alpha-numeric names, use "No." in front of the number if "No." appears at least once; c. if name strings contain Arabic and Roman numbers, use Roman number; d. strip off leading F. V.
YBrIMO	for multiple year built values, chose the one that also in Lloyds record, this rule overwrites YBrMF.
rABA	if two values in pattern of a*b*a*, then a is adopted.
rConv	if two values are within 5% difference by unit converting to metric system, then use the value in metric unit.
rDCI	decimals round to nearest integer.
rIMO	for multiple values, chose the one that is also in Lloyds record, this rule overwrites rMF.
rMAN	manual fix.

rMF	for multiple values, chose the one that occurs most frequently.
rMin	for multiple values use the lowest value, eg. vessel_key.
rNEW	manually added new value.
rNMF	if absolute difference between two numbers are within 5%, then choose the most frequent one.
rNUImo	populate a null attribute with Lloyds record.
rNUP	populate nulls with values in the following patterns: a. NNNv to vvvv; b. vNNv to vvvv; c. vNNvNu to vvvvNu; d. vNNuNuzNNz to vNNuuuzzzz. Where ‘N’ represents a null value.
rNUT	for a null/unknown value, if there's a value in the same time period, then use this value.
rONE	If the difference between two integers is 1 then merge to the most frequent one. (All decimal values were first rounded to the nearest integer.)
rSIM	If both strings are very similar due to misspelling then assess occurring frequency of each value, and adopt the most frequent one.
rSTU	set string to upper case.

Appendix 3 Final Manual Grooming Procedures

Subsequent to applying the rules in appendix 2 the following manual grooming was carried out.

Processing of data where IMO numbers were present and the Lloyd's Register could be referred to

The Lloyd's Register was referred to in all cases where IMO numbers existed and different values occurred for a single vid. These data represented about 70% of the final ungroomed dataset.

The Lloyd's Register was of limited use for call sign and registration type because these fields were often null. Only when the discrepancies were obvious were changes made. For example, on one occasion a vessel had been given the call sign '7 Jun' (which appears to be part of a date and not a call sign) and this was changed to the call sign recorded in the Lloyd's Register. It is recognised that this approach does not fully address the issue of differences in call sign and registration type.

The notation used to document amendments using the Lloyd's Register is an extension of that listed in Appendix 2, and includes the following prefixes:

Vessel name (VN)
Callsign (CS)
IMO number (IM)
Nation (NT)
DOB (YB)
OA Length (OL)
Breadth (BR)
Draught (DR)
Gross tonnage (GT)
Kilowatts (KW)
Engine make (EM)

1. RULE: - For static attributes, such as year built

   Where different values occurred for a single vid and one of these values agreed with what was recorded in the Lloyd's Register then this value was assumed to be the correct one. The notation **YBrIMO** was used to document this change.

2. RULE: - For non-static attributes, such as vessel name, overall length, kilowatts

   The same approach was taken for non-static attributes as for static attributes outlined above in point 1.
   However, the history table in the Lloyd's Register was manually accessed in all cases to check for changes in attributes, and/or conversions. If it was unclear whether the difference was legitimate, then it was assumed that it was a legitimate change of a value in *t_vessel*. The notation **VNrIMO|OLrIMO|KWrIMO**, was used to document these changes.

Processing of data where IMO numbers were not present and the Lloyd's Register could not be referred to

The frequency (count) of each attribute was examined in cases where no IMO numbers existed and different values occurred for a single vid. These data represented about 30% of the final ungroomed dataset.

For example, if a vessels breadth was 5m on 20 occasions, and 7m on 2 occasions then it was assumed that 5m was the correct breadth. It would be unusual for a vessel to be widened.
The notation used to document such amendments was **BRrMF** (Breadth changed to most frequent value).

3. RULE: - Most frequent value

It was considered that up to a 10% difference in values, between a given numeric attribute, was an acceptable level to adopt this "most frequent" approach.

However, if differences of more than 10% existed for a specific attribute, but a clear pattern was evident, it was also considered appropriate to adopt a single value. For example, if the kilowatts were:

Count	Kilowatts	Engine Make
10	2800	Caterpillar 3142
8	1750	Caterpillar 3142
9	2800	Caterpillar 3142

In this case 1750 would be changed to 2800 because there is no evidence that Engine make/model has changed. However, if the Engine make attribute was null then it would not have been possible to conclude whether 1750 was legitimate or not. Therefore, the value for kilowatts would not have been changed.

4. RULE: - Null values

In a few cases the null rules were used. These rules include:

rNUT- If the value is null/unknown and there is a value in the same time period, then use this value.

rNUP – nulls were populated with values in the following patterns: a. vNNv to vvvv; b. vNNvNu to vvvvNu; c. vNNuNuzNNz to vNNuuuzzzz. Where 'N' represents a null value

Checking of range constraints

Following the grooming process each attribute was examined to identify improbable values. For example, Breadth was restricted to the range 1-24m; Draught to the range 1-12m. There were a number of instances where these attributes had very large improbable values, however it was generally easy to determine what the error was. For example, in one instance a breadth of 103m was due to the fact that the overall length had been placed in the wrong column.

Checking of a data subset

Approximately 10% (i.e., 150) of the vessels were checked visually against the *t_vessel_org* and *t_history* tables. This was to ensure that the grooming process had worked appropriately. No obvious discrepancies were apparent.

General comments

The two main attributes that required grooming were gross tonnage and kilowatts. Gross tonnage was easier to groom than kilowatts. End users of the data should be aware that these two attributes are likely to be the least robust in this dataset because they were often null or inconsistent in the Lloyds dataset. We are confident that all other attributes, including year built, overall length, breadth, and draught are reasonably robust.