GAP Production Data Documentation

Emily Markowitz

Zack Oyafuso

Sarah Friedman

Table of contents

# 1. Welcome

Please consider this resource to be a **Living Document**. The code in this repository is regularly being updated and improved. Please refer to [releases](https://github.com/afsc-gap-products/gap_products/releases) for finalized products and project milestones.

## 1.1 What is the research objective?

The objectives of these surveys are to:

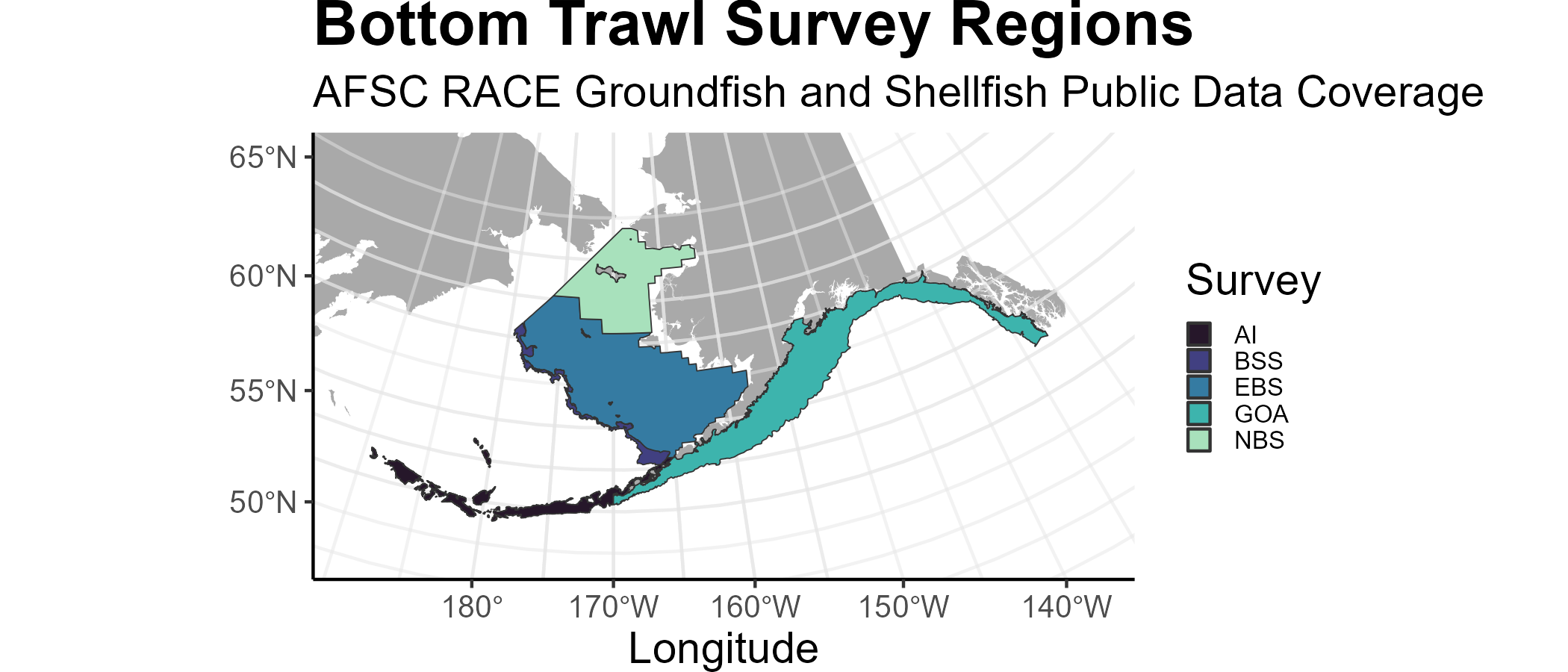
* monitor trends in the marine ecosystem of the Bering Sea, Aleutian Islands, and Gulf of Alaska,
* produce fishery-independent biomass and abundance estimates for commercially important fish and crab species, and
* collect other biological and environmental data for use in ecosystem-based fishery management.

Learn more about the [program](https://www.fisheries.noaa.gov/alaska/science-data/groundfish-assessment-program-bottom-trawl-surveys)

|  |
| --- |
| Sorting and weighing fish on deck on the 2022 Bering Sea groundfish survey aboard the F/V Alaska Knight. Credit: Emily Markowitz/NOAA Fisheries. |

# 2. Survey Background

## 2.1 Bottom trawl surveys and regions



* **Aleutian Islands (AI)** (Von Szalay and Raring, 2020)
  + Triennial (1990s)/Biennial since 2000 in even years
  + Modified Index-Stratified Random of Successful Stations Survey Design
* **Eastern Bering Sea Slope (BSS)** (Hoff, 2016)
  + Intermittent (funding dependent)
  + Modified Index-Stratified Random of Successful Stations Survey Design
* **Eastern Bering Sea Shelf (EBS)** (Markowitz et al., 2023)
  + Annual
  + Fixed stations at center of 20 x 20 nm grid
* **Gulf of Alaska (GOA)** (Von Szalay and Raring, 2018)
  + Triennial (1990s)/Biennial since 2001 in odd years
  + Stratified Random Survey Design
* **Northern Bering Sea (NBS)** (Markowitz et al., 2023)
  + Biennial/Annual
  + Fixed stations at center of 20 x 20 nm grid

| **Survey** | **Survey Definition ID** | **Years** | **Depth (m)** | **Area (km2)** | **# Statistical Areas** | **# Possible Stations** |
| --- | --- | --- | --- | --- | --- | --- |
| Aleutian Islands Bottom Trawl Survey | 52 | 2022 - 1980 (16) | 1 - 500 | 64,415.0 | 80 | 1,312 |
| Eastern Bering Sea Slope Bottom Trawl Survey | 78 | 2016 - 2002 (6) | 201 - 800 | 21,134.2 | 4 |  |
| Eastern Bering Sea Crab/Groundfish Bottom Trawl Survey | 98 | 2023 - 1982 (41) | 1 - 200 | 492,989.9 | 29 | 515 |
| Gulf of Alaska Bottom Trawl Survey | 47 | 2023 - 1984 (18) | 1 - 1,000 | 314,087.4 | 39 | 6,939 |
| Northern Bering Sea Crab/Groundfish Survey - Eastern Bering Sea Shelf Survey Extension | 143 | 2022 - 2010 (5) | 1 - 100 | 198,866.8 | 4 | 144 |

## 2.2 Survey History

### 2.2.1 Aleutian Islands Survey

### 2.2.2 Bering Sea Survey

### 2.2.3 Bering Sea Slope Survey

### 2.2.4 Gulf of Alaska Survey

# 3. Workflow

## 3.1 Workflow

**Info incoming!**

## 3.2 Data levels

GAP produces numerous data products\* that are subjected to different levels of processing, ranging from raw to highly-derived. The suitability of these data products for analysis varies and there is ambiguity about which data products can be used for which purpose. This ambiguity can create challenges in communicating about data products and potentially lead to misunderstanding and misuse of data. One approach to communicating about the level of processing applied to data products and their suitability for analysis is to describe data products using a Data Processing Level system. Data Processing Level systems are widely used in earth system sciences to characterize the extent of processing that has been applied to data products. For example, the NOAA National Centers for Environmental Information (NCEI) Satellite Program uses a Data Processing Level system to describe data on a scale of 0-4, where Level 0 is raw data and Level 4 is model output or results from analysis. Example of how [NASA remote sensing data products](https://ladsweb.modaps.eosdis.nasa.gov/search/) are shared through a public data portal with levels of data processing and documentation.

For more information, see [Sean Rohan’s October 2022 SCRUGS presentation](https://docs.google.com/presentation/d/1rWSZpeghWJqzWMIa5oBc4BCoy-zy1Yue86RoTw58u6M/edit?usp=sharing) on the topic.

* **Level 0**: Raw and unprocessed data. Ex: Data on the G drive, some tables in RACE\_DATA
* **Level 1A**: Data products with QA/QC applied that may or may not be expanded to analysis units, but either not georeferenced or does not include full metadata. Ex: Some tables in RACE\_DATA and RACEBASE
* **Level 2**: Analysis-ready data products that are derived for a standardized extent and account for zeros and missing/bad data. Ex: CPUE tables, some data products in public-facing archives and repositories
* **Level 3**: Data products that are synthesized across a standardized extent, often inputs in a higher-level analytical product. Ex: Abundance indices, some data products in public-facing archives and repositories
* **Level 4**: Analytically generated data products that are derived from lower-level data, often to inform management. Ex: Biological reference points from stock assessments, Essential Fish Habitat layers, indicators in Ecosystem Status Reports and Ecosystem and Socioeconomic Profiles

# 4. News

## 4.1 Early 2023

The main goal here to simplify the data management or to also standardize the way stock assessors are using RACE data.

We have decided to undergo this organizational change to meet the following best practices and long-term data goals. Let us know how we can better meet these objectives and best work with IT:

* Minimize duplication (both in tables and in columns within tables)
* Minimize schemata and Oracle objects to the extent possible
* Streamlined integration of tables
* Minimize work for data creators
* Minimize confusion and obstacles for data users
* Security and data management best practices

After the 2023 field season, we will deprecate the old AKFIN tables and completely replace the current tables with new tables, outlined in this document.

# 5. GAP Production Data Metadata

## 5.1 Data created in this repo

### 5.1.1 GAP\_PRODUCTS.AGECOMP

Number of rows: 544228

Number of columns: 9

SURVEY\_DEFINITION\_ID

AREA\_ID

YEAR

SPECIES\_CODE

SEX

AGE

POPULATION\_COUNT

LENGTH\_MM\_MEAN

LENGTH\_MM\_SD

52

99904

2002

10112

1

-9

11597298

480.26

121.74

52

99904

2002

10112

2

-9

11920389

597.30

149.62

52

99904

2004

10112

1

-9

17929130

433.52

155.49

### 5.1.2 GAP\_PRODUCTS.AREA

This reference table stores all metadata and estimates for all estimates of stratum and subarea area estimates. Use this table with the STRATUM\_GROUPS and SURVEY\_DESIGN tables. by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap\_products. For more information about codes used in the tables, please refer to the survey code books (https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). These data were last updated June 27, 2023.

Number of rows: 443

Number of columns: 10

SURVEY\_DEFINITION\_ID

DESIGN\_YEAR

AREA\_ID

TYPE

AREA\_NAME

DESCRIPTION

AREA\_KM2

DEPTH\_MIN\_M

DEPTH\_MAX\_M

crs

98

2022

62

STRATUM

Outer Domain

100-200 m, NW section, high density, St. Matthew - OUTER DOMAIN

6461.505

101

200

NA

143

2022

70

STRATUM

Inner Domain

<50 m , N of standard area, N to St.Lawrence Is.

79259.889

1

50

NA

143

2022

71

STRATUM

Inner Domain

<50 m, Norton Sound and N of St. Lawrence Island to Bering Strait. Omits AA-10

81255.025

1

50

NA

### 5.1.3 GAP\_PRODUCTS.BIOMASS

Number of rows: 4582456

Number of columns: 16

SURVEY\_DEFINITION\_ID

AREA\_ID

SPECIES\_CODE

YEAR

N\_HAUL

N\_WEIGHT

N\_COUNT

N\_LENGTH

CPUE\_KGKM2\_MEAN

CPUE\_KGKM2\_VAR

CPUE\_NOKM2\_MEAN

CPUE\_NOKM2\_VAR

BIOMASS\_MT

BIOMASS\_VAR

POPULATION\_COUNT

POPULATION\_VAR

52

211

24193

1991

5

0

0

0

0

0

0

0

0

0

0

0

52

211

24195

1991

5

0

0

0

0

0

0

0

0

0

0

0

52

211

24220

1991

5

0

0

0

0

0

0

0

0

0

0

0

### 5.1.4 GAP\_PRODUCTS.CPUE

Haul-level zero-filled weight and numerical catch-per-unit-effort. This table was created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap\_products. For more information about codes used in the tables, please refer to the survey code books (https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). These data were last updated September 03, 2023.

Number of rows: 37655036

Number of columns: 7

HAULJOIN

SPECIES\_CODE

WEIGHT\_KG

COUNT

AREA\_SWEPT\_KM2

CPUE\_KGKM2

CPUE\_NOKM2

-21974

44102

0

0

0.026936

0

0

-21974

44103

0

0

0.026936

0

0

-21974

44104

0

0

0.026936

0

0

### 5.1.5 GAP\_PRODUCTS.DESIGN\_SURVEY

Number of rows: 42S02 942 [Oracle][ODBC][Ora]ORA-00942: table or view does not exist

Number of columns:

x

42S02 942 [Oracle][ODBC][Ora]ORA-00942: table or view does not exist

[RODBC] ERROR: Could not SQLExecDirect ’SELECT \* FROM GAP\_PRODUCTS.DESIGN\_SURVEY FETCH FIRST 3 ROWS ONLY;’

### 5.1.6 GAP\_PRODUCTS.METADATA\_TABLE

Number of rows: 8

Number of columns: 3

METADATA\_SENTENCE\_NAME

METADATA\_SENTENCE\_TYPE

METADATA\_SENTENCE

survey\_institution

fragment

by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC).

legal\_restrict

sentence

There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries).

legal\_restrict\_none

sentence

There are no legal restrictions on access to the data.

### 5.1.7 GAP\_PRODUCTS.STRATUM\_GROUPS

This is a table

Number of rows: 774

Number of columns: 4

AREA\_ID

SURVEY\_DEFINITION\_ID

DESIGN\_YEAR

STRATUM

1

98

2022

10

1

78

2002

11

1

78

2002

12

### 5.1.8 GAP\_PRODUCTS.SIZECOMP

Number of rows: 3113209

Number of columns: 7

SURVEY\_DEFINITION\_ID

YEAR

AREA\_ID

SPECIES\_CODE

LENGTH\_MM

SEX

POPULATION\_COUNT

52

2002

291

10110

150

2

9342

52

2002

291

10110

170

2

9946

52

2002

291

10110

180

2

41639

, ### GAP\_PRODUCTS.AGECOMP

Number of rows: 544228

Number of columns: 9

SURVEY\_DEFINITION\_ID

AREA\_ID

YEAR

SPECIES\_CODE

SEX

AGE

POPULATION\_COUNT

LENGTH\_MM\_MEAN

LENGTH\_MM\_SD

52

99904

2002

10112

1

-9

11597298

480.26

121.74

52

99904

2002

10112

2

-9

11920389

597.30

149.62

52

99904

2004

10112

1

-9

17929130

433.52

155.49

### 5.1.9 GAP\_PRODUCTS.AREA

This reference table stores all metadata and estimates for all estimates of stratum and subarea area estimates. Use this table with the STRATUM\_GROUPS and SURVEY\_DESIGN tables. by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap\_products. For more information about codes used in the tables, please refer to the survey code books (https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). These data were last updated June 27, 2023.

Number of rows: 443

Number of columns: 10

SURVEY\_DEFINITION\_ID

DESIGN\_YEAR

AREA\_ID

TYPE

AREA\_NAME

DESCRIPTION

AREA\_KM2

DEPTH\_MIN\_M

DEPTH\_MAX\_M

crs

98

2022

62

STRATUM

Outer Domain

100-200 m, NW section, high density, St. Matthew - OUTER DOMAIN

6461.505

101

200

NA

143

2022

70

STRATUM

Inner Domain

<50 m , N of standard area, N to St.Lawrence Is.

79259.889

1

50

NA

143

2022

71

STRATUM

Inner Domain

<50 m, Norton Sound and N of St. Lawrence Island to Bering Strait. Omits AA-10

81255.025

1

50

NA

### 5.1.10 GAP\_PRODUCTS.BIOMASS

Number of rows: 4582456

Number of columns: 16

SURVEY\_DEFINITION\_ID

AREA\_ID

SPECIES\_CODE

YEAR

N\_HAUL

N\_WEIGHT

N\_COUNT

N\_LENGTH

CPUE\_KGKM2\_MEAN

CPUE\_KGKM2\_VAR

CPUE\_NOKM2\_MEAN

CPUE\_NOKM2\_VAR

BIOMASS\_MT

BIOMASS\_VAR

POPULATION\_COUNT

POPULATION\_VAR

52

211

24193

1991

5

0

0

0

0

0

0

0

0

0

0

0

52

211

24195

1991

5

0

0

0

0

0

0

0

0

0

0

0

52

211

24220

1991

5

0

0

0

0

0

0

0

0

0

0

0

### 5.1.11 GAP\_PRODUCTS.CPUE

Haul-level zero-filled weight and numerical catch-per-unit-effort. This table was created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap\_products. For more information about codes used in the tables, please refer to the survey code books (https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). These data were last updated September 03, 2023.

Number of rows: 37655036

Number of columns: 7

HAULJOIN

SPECIES\_CODE

WEIGHT\_KG

COUNT

AREA\_SWEPT\_KM2

CPUE\_KGKM2

CPUE\_NOKM2

-21974

44102

0

0

0.026936

0

0

-21974

44103

0

0

0.026936

0

0

-21974

44104

0

0

0.026936

0

0

### 5.1.12 GAP\_PRODUCTS.DESIGN\_SURVEY

Number of rows: [RODBC] ERROR: Could not SQLExecDirect ’SELECT COUNT(\*) FROM GAP\_PRODUCTS.DESIGN\_SURVEY;’

Number of columns:

x

42S02 942 [Oracle][ODBC][Ora]ORA-00942: table or view does not exist

[RODBC] ERROR: Could not SQLExecDirect ’SELECT \* FROM GAP\_PRODUCTS.DESIGN\_SURVEY FETCH FIRST 3 ROWS ONLY;’

### 5.1.13 GAP\_PRODUCTS.METADATA\_TABLE

Number of rows: 8

Number of columns: 3

METADATA\_SENTENCE\_NAME

METADATA\_SENTENCE\_TYPE

METADATA\_SENTENCE

survey\_institution

fragment

by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC).

legal\_restrict

sentence

There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries).

legal\_restrict\_none

sentence

There are no legal restrictions on access to the data.

### 5.1.14 GAP\_PRODUCTS.STRATUM\_GROUPS

This is a table

Number of rows: 774

Number of columns: 4

AREA\_ID

SURVEY\_DEFINITION\_ID

DESIGN\_YEAR

STRATUM

1

98

2022

10

1

78

2002

11

1

78

2002

12

### 5.1.15 GAP\_PRODUCTS.SIZECOMP

Number of rows: 3113209

Number of columns: 7

SURVEY\_DEFINITION\_ID

YEAR

AREA\_ID

SPECIES\_CODE

LENGTH\_MM

SEX

POPULATION\_COUNT

52

2002

291

10110

150

2

9342

52

2002

291

10110

170

2

9946

52

2002

291

10110

180

2

41639

# 6. Universal Column Metadata

These tables provide the column metadata for all of the tables and views in GAP\_PRODUCTS. These tables are created by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap\_products. These data were last updated August 08, 2023. There are no legal restrictions on access to the data. For more information about codes used in the tables, please refer to the survey code books (https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual).

| **Column name from data** | **Descriptive column Name** | **Units** | **Oracle data type** | **Column description** |
| --- | --- | --- | --- | --- |
| ABUNDANCE\_HAUL | Design-based index approved haul | logical |  | Logical, describing if this haul was conducted in a standard manner and thus used for design-based index estimates (TRUE) or not (FALSE). |
| ACTIVE | Vessel Active/Inactive | logical |  | Logical, describing if a vessel is active (TRUE) or not (FALSE). |
| AGE | Age bin of taxon | year | NUMBER(38,0) | Age bin of a taxon in years estimated by the age comp estimate. |
| AGENCY\_ACRONYM | Acroynm of listed Agency | text abbreviated | VARCHAR2(255 BYTE) | Abbreviated agencies that are affiliated with the Alaska bottom trawl survey. The column 'agency\_acronym' is associated with the 'agency\_short' and 'agency\_long' columns. |
| AGENCY\_JOIN | Agency's ID code | ID code | NUMBER(38,0) | Affiliated agency ID code. |
| AGENCY\_LONG | Agency's Offical Name | text | VARCHAR2(255 BYTE) | Full official name of affiliated agencies to the Alaska bottom trawl survey. The column 'agency\_long' is associated with the 'agency\_acronym' and 'agency\_short' columns. |
| AGENCY\_SHORT | Agency's Shorthand Name | text | VARCHAR2(255 BYTE) | A sort version of the full official name of affiliated agencies to the Alaska bottom trawl survey. The column 'agency\_short' is associated with the 'agency\_acronym' and 'agency\_long' columns. |
| AREA\_ID | Area ID code | ID code | NUMBER(38,0) | Area ID code for each statistical area used to produce production estimates (e.g., biomass, population, age comps, length comps). Each area ID is unique within each survey. |
| AREA\_KM2 | Area (km<sup>2</sup>) | kilometers squared | NUMBER(38,3) | Area in thousands of square kilometers. |
| AREA\_NAME | Area ID Name | text | VARCHAR2(4000 BYTE) | Descriptive name of each AREA\_ID. These names often identify the region, depth ranges, or other regional information for the area ID. |
| AREA\_SWEPT\_KM2 | Area Swept (km) | kilometers | NUMBER(38,6) | The area the net covered while the net was fishing (kilometers squared), defined as the distance fished times the net width. |
| AREA\_TYPE | Area ID Type Description | category | VARCHAR2(255 BYTE) | The type of stratum that AREA\_ID represents. Types include: STRATUM, REGION, DEPTH, SUBAREA, INPFC BY DEPTH, INPFC, SUBAREA BY DEPTH, REGULATORY AREA, NMFS STATISTICAL AREA. |
| BIOMASS\_CI\_LOWER | Estimated Biomass Lower Confidence Interval | numeric | NUMBER(38,6) | The estimated biomass lower confidence interval caught in the survey for a species, group, or total for a given survey. |
| BIOMASS\_CI\_UPPER | Estimated Biomass Upper Confidence Interval | numeric | NUMBER(38,6) | The estimated biomass upper confidence interval caught in the survey for a species, group, or total for a given survey. |
| BIOMASS\_DF | Estimated Biomass Degrees of Freedom | numeric | NUMBER(38,6) | The estimated biomass degrees of freedom caught in the survey for a species, group, or total for a given survey. |
| BIOMASS\_MT | Estimated Biomass | numeric | NUMBER(38,6) | The estimated biomass caught in the survey for a species, group, or total for a given survey. |
| BIOMASS\_VAR | Estimated Biomass Variance | numeric | NUMBER(38,6) | The estimated biomass variance caught in the survey for a species, group, or total for a given survey. |
| BOTTOM\_TEMPERATURE\_C | Bottom Temperature (Degrees Celsius) | degrees Celsius | NUMBER(38,1) | Bottom temperature (tenths of a degree Celsius); NA indicates removed or missing values. |
| BOTTOM\_TYPE\_CODE | Seafloor bottom type code | ID code | NUMBER(38,0) | Bottom type on sea floor at haul location. For a complete list of bottom type ID codes, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| CLASSIFICATION | Taxonomic classification rank group | category | VARCHAR2(255 BYTE) | Phylogenetic classification group rank for a given species. |
| CLASS\_TAXON | Class phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of class\_taxon of a given species. |
| COMMENTS | Comments | text | VARCHAR2(4000 BYTE) | Comments regarding row observation. |
| COMMON\_NAME | Taxon Common Name | text | VARCHAR2(255 BYTE) | The common name of the marine organism associated with the 'scientific\_name' and 'species\_code' columns. For a complete species list, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| COUNT | Taxon Count | count, whole number resolution | NUMBER(38,0) | Total number of individuals caught in haul by taxon, represented in whole numbers used in calculation. |
| COUNTRY\_ID | Vessel Name | text | VARCHAR2(255 BYTE) | Name of the vessel used to collect data for that haul. The column 'vessel\_name' is associated with the 'vessel\_id' column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| CPUE\_KGHA | Weight CPUE (kg/ha) | kilograms per hectare | NUMBER(38,6) | Catch weight (kilograms) divided by area (hectares) swept by the net. |
| CPUE\_KGKM2 | Weight CPUE (kg/km<sup>2</sup>) | kilograms per kilometers squared | NUMBER(38,6) | Catch weight (kilograms) divided by area (squared kilometers) swept by the net. |
| CPUE\_KGKM2\_MEAN | Mean Weight CPUE | kilograms per kilometers squared | NUMBER(38,6) | The mean of catch weight (kilograms) divided by area (squared kilometers) swept by the net used in design-based indicie calculation. |
| CPUE\_KGKM2\_VAR | Variance of the Mean Weight CPUE | kilograms per kilometers squared | NUMBER(38,6) | The variance of mean of catch weight (kilograms) divided by area (squared kilometers) swept by the net used in design-based indicie calculation. |
| CPUE\_NOHA | Number CPUE (no/ha) | count per hectare | NUMBER(38,6) | Catch number (in number of organisms) per area (hectares) swept by the net. |
| CPUE\_NOKM2 | Number CPUE (no/km<sup>2</sup>) | count per kilometers squared | NUMBER(38,6) | Catch number (in number of organisms) per area (squared kilometers) swept by the net. |
| CPUE\_NOKM2\_MEAN | Mean Numberic CPUE | count per kilometers squared | NUMBER(38,6) | The mean of catch count (number) divided by area (squared kilometers) swept by the net used in design-based indicie calculation. |
| CPUE\_NOKM2\_VAR | Variance of the Mean Numeric CPUE | count per kilometers squared | NUMBER(38,6) | The variance of mMean of catch count (number) divided by area (squared kilometers) swept by the net used in design-based indicie calculation. |
| CRS | Coordinate Reference System | ID code | VARCHAR2(5 BYTE) | Coordinate reference system that areas (like AREA\_KM2) are calculated in, as defined by https://spatialreference.org/ (e.g., "+proj=longlat", "EPSG:3338"). |
| CRUISE | Cruise ID | ID code | NUMBER(38,0) | This is a six-digit number identifying the cruise number of the form: YYYY99 (where YYYY = year of the cruise; 99 = 2-digit number and is sequential; 01 denotes the first cruise that vessel made in this year, 02 is the second, etc.). |
| CRUISEJOIN | Cruise ID | ID code | NUMBER(38,0) | This is a unique numeric identifier assigned to each survey, vessel, and year combination. |
| DATABASE | Genus phylogenetic rank | category | VARCHAR2(255 BYTE) | Taxonomic database source (e.g., "ITIS", "WORMS"). |
| DATABASE\_ID | Subfamily phylogenetic rank | ID code | VARCHAR2(255 BYTE) | Species ID code of a species in the taxonomic "DATABASE" source. |
| DATE | Date | YYYY-MM-DD | DATE | The date (YYYY-MM-DD) of the event (e.g., cruise). |
| DATE\_END | End Date | YYYY-MM-DD | DATE | The date (YYYY-MM-DD) of the end of the event (e.g., cruise). |
| DATE\_START | Start Date | YYYY-MM-DD | DATE | The date (YYYY-MM-DD) of the beginning of the event (e.g., cruise). |
| DATE\_TIME | Date and Time | MM/DD/YYYY HH::MM | DATE | The date (MM/DD/YYYY) and time (HH:MM) of the haul. |
| DATE\_TIME\_END | End Date and Time | MM/DD/YYYY HH::MM | DATE | The date (MM/DD/YYYY) and time (HH:MM) of the end of the haul. |
| DATE\_TIME\_START | Start Date and Time | MM/DD/YYYY HH::MM | DATE | The date (MM/DD/YYYY) and time (HH:MM) of the beginning of the haul. |
| DEPTH\_M | Depth (m) | degrees Celsius | NUMBER(38,1) | Bottom depth (tenths of a meter). |
| DEPTH\_MAX\_M | Area ID Maximum Depth (m) | meters | NUMBER(38,3) | Maximum depth (meters) of the area covered by AREA\_ID. |
| DEPTH\_MIN\_M | Area ID Minimum Depth (m) | meters | NUMBER(38,3) | Minimum depth (meters) of the area covered by AREA\_ID. |
| DESCRIPTION | Description | text | VARCHAR2(4000 BYTE) | Description of row observation. |
| DESIGN\_YEAR | Design year | year | NUMBER(10,0) | The year the survey area stratum (e.g., statistical stratum, summary area, region) was implimented in. |
| DISTANCE\_FISHED\_KM | Distance Fished (km) | degrees Celsius | NUMBER(38,3) | Distance the net fished (thousandths of kilometers). |
| DUMMY | dummy | dummy | VARCHAR2(255 BYTE) | dummy |
| DURATION\_HR | Tow Duration (decimal hr) | hours | NUMBER(38,1) | This is the elapsed time between start and end of a haul (decimal hours). |
| FAMILY | Suborder phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of family of a given species. |
| GEAR\_DEPTH\_M | Gear depth | meters | NUMBER(38,1) | Depth gear was deployed at (tenths of a meter). Gear depth plus net height equals bottom depth. |
| GEAR\_ID | Gear ID code | ID code | NUMBER(38,0) | Type of trawl or gear deployed. For a complete list of vessel gear type ID codes, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| GENUS\_TAXON | Genus phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of genus\_taxon of a given species. |
| HAUL | Haul Number | ID code | NUMBER(38,0) | This number uniquely identifies a sampling event (haul) within a cruise. It is a sequential number, in chronological order of occurrence. |
| HAULJOIN | Haul ID | ID code | NUMBER(38,0) | This is a unique numeric identifier assigned to each (vessel, cruise, and haul) combination. |
| HAUL\_TYPE | Haul Sampling Type | ID code | NUMBER(38,0) | Type of haul sampling method. For a complete list of haul type ID codes, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| ID\_RANK | Lowest taxonomic rank | text | VARCHAR2(255 BYTE) | Lowest taxonomic rank of a given species entry. |
| ITIS | ITIS Taxonomic Serial Number | ID code | NUMBER(38,0) | Species code as identified in the Integrated Taxonomic Information System (https://itis.gov/). |
| KINGDOM\_TAXON | kingdom\_taxon phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of kingdom\_taxon of a given species. |
| LATITUDE\_DD | Latitude (decimal degrees) | decimal degrees | NUMBER(38,6) | Latitude (one hundred thousandth of a decimal degree). |
| LATITUDE\_DD\_END | End Latitude (decimal degrees) | decimal degrees | NUMBER(38,6) | Latitude (one hundred thousandth of a decimal degree) of the end of the haul. |
| LATITUDE\_DD\_START | Start Latitude (decimal degrees) | decimal degrees | NUMBER(38,6) | Latitude (one hundred thousandth of a decimal degree) of the start of the haul. |
| LENGTH\_MM | Length of a specimen | millimeters | NUMBER(10,0) | Length of a specimen in millimeters. |
| LENGTH\_MM\_MEAN | Mean length at age weighted by numbers at length | numeric | NUMBER(38,3) | Mean length estimated in age comp estimate. |
| LENGTH\_MM\_SD | standard deviation of length at age weighted by numbers at length | numeric | NUMBER(38,3) | Variance of mean length estimated in age comp estimate. |
| LONGITUDE\_DD | Longitude (decimal degrees) | decimal degrees | NUMBER(38,6) | Longitude (one hundred thousandth of a decimal degree). |
| LONGITUDE\_DD\_END | End Longitude (decimal degrees) | decimal degrees | NUMBER(38,6) | Longitude (one hundred thousandth of a decimal degree) of the end of the haul. |
| LONGITUDE\_DD\_START | Start Longitude (decimal degrees) | decimal degrees | NUMBER(38,6) | Longitude (one hundred thousandth of a decimal degree) of the start of the haul. |
| METADATA\_COLNAME | Column name | text | VARCHAR2(255 BYTE) | Name of the column in a table. |
| METADATA\_COLNAME\_DESC | column description | text | VARCHAR2(4000 BYTE) | Descritpion of the column. |
| METADATA\_COLNAME\_LONG | Column name spelled out | text | VARCHAR2(255 BYTE) | Long name for the column. |
| METADATA\_SENTENCE | Sentence | text | VARCHAR2(255 BYTE) | Table metadata sentence. |
| METADATA\_SENTENCE\_NAME | Metadata sentence name | text | VARCHAR2(255 BYTE) | Name of table metadata sentence. |
| METADATA\_SENTENCE\_TYPE | Sentence type | text | VARCHAR2(255 BYTE) | Type of sentence to have in table metadata. |
| METADATA\_UNITS | Units | category | VARCHAR2(255 BYTE) | Units of the column. |
| NET\_HEIGHT\_M | Net Height (m) | meters | NUMBER(38,1) | Measured or estimated distance (meters) between footrope and headrope of the trawl. |
| NET\_MEASURED | Net measured during haul | logical |  | Logical, describing if the net was measured (TRUE) or not (FALSE) by wheelhouse and marport programs during the haul. |
| NET\_WIDTH\_M | Net Width (m) | meters | NUMBER(38,1) | Measured or estimated distance (meters) between wingtips of the trawl. |
| N\_COUNT | Hauls with taxon counts | numeric | NUMBER(38,0) | Total number of hauls with positive taxon counts used in calculation. |
| N\_HAUL | Valid hauls | numeric | NUMBER(38,0) | Total number of valid hauls used in calculation. |
| N\_LENGTH | Hauls with taxon lengths | numeric | NUMBER(38,0) | Total number of hauls with taxon length data used in calculation. |
| N\_WEIGHT | Hauls with catch | numeric | NUMBER(38,0) | Total number of hauls with positive catch/weighed taxon data used in calculation. |
| ORDER\_TAXON | Subclass phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of order\_taxon of a given species. |
| PERFORMANCE | Haul Performance Code | category | NUMBER(38,0) | This denotes what, if any, issues arose during the haul. For more information, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| PHYLUM\_TAXON | phylum\_taxon phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of phylum\_taxon of a given species. |
| POPULATION\_CI\_LOWER | Estimated Population Lower Confidence Interval | numeric | NUMBER(38,6) | The estimated population lower confidence interval caught in the survey for a species, group, or total for a given survey. |
| POPULATION\_CI\_UPPER | Estimated Population Upper Confidence Interval | numeric | NUMBER(38,6) | The estimated population upper confidence interval caught in the survey for a species, group, or total for a given survey. |
| POPULATION\_COUNT | Estimated Population | numeric | NUMBER(38,6) | The estimated population caught in the survey for a species, group, or total for a given survey. |
| POPULATION\_DF | Estimated Population Degrees of Freedom | numeric | NUMBER(38,6) | The estimated population degrees of freedom caught in the survey for a species, group, or total for a given survey. |
| POPULATION\_VAR | Estimated Population Variance | numeric | NUMBER(38,6) | The estimated population variance caught in the survey for a species, group, or total for a given survey. |
| PRINCIPAL\_INVESTIGATOR | Principle Investigator | text | VARCHAR2(255 BYTE) | First and last name of principle investigator for a project. |
| PROJECT\_TITLE | Title of Special Project | text | VARCHAR2(255 BYTE) | Special project title. |
| PROJECT\_TITLE\_SHORT | Short Title of Special Project | text | VARCHAR2(255 BYTE) | Special project short title (short version of PROJECT\_TITLE). |
| REASON | Naming status for species | text | VARCHAR2(5 BYTE) | Description of species' naming status (e.g., "synonym", "preoccupied","misspelling" ) |
| SCIENTIFIC\_NAME | Taxon Scientific Name | text | VARCHAR2(255 BYTE) | The scientific name of the organism associated with the 'common\_name' and 'species\_code' columns. For a complete taxon list, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| SEX | Sex of a specimen | ID code | NUMBER(38,0) | Sex of a specimen where "1" = "Male", "2" = "Female", "3" = Unsexed. |
| SPECIES\_CODE | Taxon Code | ID code | NUMBER(38,0) | The species code of the organism associated with the 'common\_name' and 'scientific\_name' columns. For a complete species list, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| SPECIES\_NAME\_ACCEPTED | Scientific name used in taxonomic database | text | VARCHAR2(5 BYTE) | Scientific name of species used in taxonomic "DATABASE" column. |
| SPECIES\_NAME\_SURVEY | Scientific name used in survey data | text | VARCHAR2(5 BYTE) | Scientific name of species historically or currently used in the survey. |
| SRVY | Survey | text abbreviated | VARCHAR2(255 BYTE) | Abbreviated survey names. The column 'srvy' is associated with the 'survey' and 'survey\_id' columns. Northern Bering Sea (NBS), Southeastern Bering Sea (EBS), Bering Sea Slope (BSS), Gulf of Alaska (GOA), Aleutian Islands (AI). |
| STATION | Station ID | ID code | VARCHAR2(255 BYTE) | Alpha-numeric designation for the station established in the design of a survey. |
| STRATUM | Stratum ID | ID code | NUMBER(10,0) | RACE database statistical area for analyzing data. Strata were designed using bathymetry and other geographic and habitat-related elements. The strata are unique to each survey series. Stratum of value 0 indicates experimental tows. |
| SUBCLASS\_TAXON | Subclass phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of subclass\_taxon of a given species. |
| SUBFAMILY\_TAXON | Subfamily phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of subfamily\_taxon of a given species. |
| SUBMISSION\_DATE | Date | YYYY-MM-DD | TIMESTAMP | Date special projects were due to be submitted for the upcoming survey season. |
| SUBORDER\_TAXON | Suborder phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of suborder\_taxon of a given species. |
| SUBPHYLUM\_TAXON | Subphylum phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of subphylum\_taxon of a given species. |
| SUPERCLASS\_TAXON | Superclass phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of superclass\_taxon of a given species. |
| SUPERFAMILY\_TAXON | Superfamily phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of superfamily\_taxon of a given species. |
| SUPERORDER | Superorder phylogenetic rank | category | VARCHAR2(255 BYTE) | Phylogenetic latin rank of superorder of a given species. |
| SURFACE\_TEMPERATURE\_C | Surface Temperature (Degrees Celsius) | degrees Celsius | NUMBER(38,1) | Surface temperature (tenths of a degree Celsius); NA indicates removed or missing values. |
| SURVEY | Survey Name | text | VARCHAR2(255 BYTE) | Name and description of survey. The column 'survey' is associated with the 'srvy' and 'survey\_id' columns. |
| SURVEY\_DEFINITION\_ID | Survey ID | ID code | NUMBER(38,0) | This number uniquely identifies a survey. Name and description of survey. The column 'survey\_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| SURVEY\_ID | Survey ID | ID code | NUMBER(38,0) | This number uniquely identifies a survey. Name and description of survey. The column 'survey\_id' is associated with the 'srvy' and 'survey' columns. For a complete list of surveys, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| TAXON\_CONFIDENCE | Taxon Confidence Rating | category | VARCHAR2(255 BYTE) | Confidence in the ability of the survey team to correctly identify the taxon to the specified level, based solely on identification skill (e.g., not likelihood of a taxon being caught at that station on a location-by-location basis). Quality codes follow: \*\*'High'\*\*: High confidence and consistency. Taxonomy is stable and reliable at this level, and field identification characteristics are well known and reliable. \*\*'Moderate'\*\*: Moderate confidence. Taxonomy may be questionable at this level, or field identification characteristics may be variable and difficult to assess consistently. \*\*'Low'\*\*: Low confidence. Taxonomy is incompletely known, or reliable field identification characteristics are unknown. Documentation: [Species identification confidence in the eastern Bering Sea shelf survey (1982-2008)](http://apps-afsc.fisheries.noaa.gov/Publications/ProcRpt/PR2009-04.pdf), [Species identification confidence in the eastern Bering Sea slope survey (1976-2010)](http://apps-afsc.fisheries.noaa.gov/Publications/ProcRpt/PR2014-05.pdf), and [Species identification confidence in the Gulf of Alaska and Aleutian Islands surveys (1980-2011)](http://apps-afsc.fisheries.noaa.gov/Publications/ProcRpt/PR2014-01.pdf). |
| TAXON\_CONFIDENCE\_CODE | Taxon Confidence Rating | category | NUMBER(38,0) | Confidence in the ability of the survey team to correctly identify the taxon to the specified level, based solely on identification skill (e.g., not likelihood of a taxon being caught at that station on a location-by-location basis). Quality codes follow: \*\*'High'\*\*: High confidence and consistency. Taxonomy is stable and reliable at this level, and field identification characteristics are well known and reliable. \*\*'Moderate'\*\*: Moderate confidence. Taxonomy may be questionable at this level, or field identification characteristics may be variable and difficult to assess consistently. \*\*'Low'\*\*: Low confidence. Taxonomy is incompletely known, or reliable field identification characteristics are unknown. Documentation: [Species identification confidence in the eastern Bering Sea shelf survey (1982-2008)](http://apps-afsc.fisheries.noaa.gov/Publications/ProcRpt/PR2009-04.pdf), [Species identification confidence in the eastern Bering Sea slope survey (1976-2010)](http://apps-afsc.fisheries.noaa.gov/Publications/ProcRpt/PR2014-05.pdf), and [Species identification confidence in the Gulf of Alaska and Aleutian Islands surveys (1980-2011)](http://apps-afsc.fisheries.noaa.gov/Publications/ProcRpt/PR2014-01.pdf). |
| TRAWLABLE | Trawlable stations | logical |  | Logical, describing if stations are trawlable (TRUE) or not (FALSE). |
| VESSEL\_CALLSIGN | Vessel Call Sign | ID code | NUMBER(38,0) | A call sign is a designated sequence of letters and numbers that are assigned when a vessel, whether it be a sailing yacht, motor yacht, rib or commercial vessel, receives it's Ship Radio Licence. The vessel also receives it's MMSI number, so that each vessel is uniquely identified. |
| VESSEL\_COAST\_GUARD\_NUMBER | Vessel Coast Guard Number | ID code | NUMBER(38,0) | Official Identification number as defined by www.dco.uscg.mil. The Official Number (O/N) is the 6 or 7 digit number awarded to the vessel at the time it is first documented with the US Coast Guard. This number remains with the vessel indefinitely and should be marked in accordance with 46 CFR 67.121. |
| VESSEL\_ID | Vessel ID | ID code | NUMBER(38,0) | ID number of the vessel used to collect data for that haul. The column 'vessel\_id' is associated with the 'vessel\_name' column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| VESSEL\_IMO | Vessel International Maritime Organization Number | ID code | NUMBER(38,0) | The International Maritime Organization (IMO) number consists of the letters "IMO" followed by a unique, seven-digit number: the pattern is "NNNNNNN", where N is a single-digit number, e.g., "1234567" |
| VESSEL\_LENGTH\_M | Vessel Length | meters | NUMBER(38,0) | The length of vessel in meters. |
| VESSEL\_MMSI | Vessel Maritime Mobile Service Identities | ID code | NUMBER(38,0) | Maritime Mobile Service Identities (MMSIs) are nine-digit numbers used by maritime digital selective calling (DSC), automatic identification systems (AIS) and certain other equipment to uniquely identify a ship or a coast radio station. |
| VESSEL\_NAME | Vessel Name | text | VARCHAR2(255 BYTE) | Name of the vessel used to collect data for that haul. The column 'vessel\_name' is associated with the 'vessel\_id' column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). |
| VESSEL\_OWNER | Vessel Owner | text | VARCHAR2(255 BYTE) | Name of vessel owner or company. |
| VESSEL\_TONNAGE | Vessel Tonnage | metric tons | NUMBER(38,0) | The tonnage of vessel in metric tons. |
| WEIGHT\_KG | Taxon Weight (kg) | kilograms | NUMBER(38,3) | Weight (thousandths of a kilogram) of individuals in a haul by taxon. |
| WIRE\_LENGTH\_M | Trawl wire length | meters | NUMBER(38,0) | Length of wire deployed during a given haul in meters. |
| WORMS | World Register of Marine Species Taxonomic Serial Number | ID code | NUMBER(38,0) | Species code as identified in the World Register of Marine Species (WoRMS) (https://www.marinespecies.org/). |
| YEAR | Year | year | NUMBER(10,0) | Year the survey was conducted in. |

# 7. GAP Production Data - Created using {gapindex}

Our production data is created using the {gapindex} R package. [Insert info and examples from {gapindex}]

# 8. AFSC RACE Groundfish Data for AKFIN Metadata

## 8.1 Data description

*In development*

## 8.2 Data created in this repo

### 8.2.1 AKFIN\_AGECOMP

Number of rows: 543727

Number of columns: 10

SURVEY

SURVEY\_DEFINITION\_ID

AREA\_ID

YEAR

SPECIES\_CODE

SEX

AGE

POPULATION\_COUNT

LENGTH\_MM\_MEAN

LENGTH\_MM\_SD

AI

52

99904

2018

10110

3

4

5978

249.02

19.98

AI

52

99904

2018

10110

3

8

974

560.00

0.00

AI

52

99904

2018

10110

3

9

2923

560.00

0.00

### 8.2.2 AKFIN\_AREA

Number of rows: 443

Number of columns: 10

SURVEY\_DEFINITION\_ID

DESIGN\_YEAR

AREA\_ID

TYPE

AREA\_NAME

DESCRIPTION

AREA\_KM2

DEPTH\_MIN\_M

DEPTH\_MAX\_M

crs

98

2022

62

STRATUM

Outer Domain

100-200 m, NW section, high density, St. Matthew - OUTER DOMAIN

6461.505

101

200

NA

143

2022

70

STRATUM

Inner Domain

<50 m , N of standard area, N to St.Lawrence Is.

79259.889

1

50

NA

143

2022

71

STRATUM

Inner Domain

<50 m, Norton Sound and N of St. Lawrence Island to Bering Strait. Omits AA-10

81255.025

1

50

NA

### 8.2.3 AKFIN\_BIOMASS

Number of rows: 4549332

Number of columns: 17

SURVEY\_DEFINITION\_ID

SURVEY

AREA\_ID

SPECIES\_CODE

YEAR

N\_HAUL

N\_WEIGHT

N\_COUNT

N\_LENGTH

CPUE\_KGKM2\_MEAN

CPUE\_KGKM2\_VAR

CPUE\_NOKM2\_MEAN

CPUE\_NOKM2\_VAR

BIOMASS\_MT

BIOMASS\_VAR

POPULATION\_COUNT

POPULATION\_VAR

52

AI

211

23861

1991

5

0

0

0

0

0

0

0

0

0

0

0

52

AI

211

23862

1991

5

0

0

0

0

0

0

0

0

0

0

0

52

AI

211

23863

1991

5

0

0

0

0

0

0

0

0

0

0

0

### 8.2.4 AKFIN\_CPUE

Number of rows: 37209741

Number of columns: 7

HAULJOIN

SPECIES\_CODE

WEIGHT\_KG

COUNT

AREA\_SWEPT\_KM2

CPUE\_KGKM2

CPUE\_NOKM2

-21974

44102

0

0

0.026936

0

0

-21974

44103

0

0

0.026936

0

0

-21974

44104

0

0

0.026936

0

0

### 8.2.5 AKFIN\_HAUL

Number of rows: 35096

Number of columns: 26

CRUISEJOIN

HAULJOIN

HAUL

HAUL\_TYPE

VESSEL\_ID

PERFORMANCE

DATE\_TIME\_START

DURATION\_HR

DISTANCE\_FISHED\_KM

NET\_WIDTH\_M

NET\_MEASURED

NET\_HEIGHT\_M

STRATUM

LATITUDE\_DD\_START

LATITUDE\_DD\_END

LONGITUDE\_DD\_START

LONGITUDE\_DD\_END

STATION

DEPTH\_GEAR\_M

DEPTH\_M

BOTTOM\_TYPE

SURFACE\_TEMPERATURE\_C

GEAR\_TEMPERATURE\_C

WIRE\_LENGTH\_M

GEAR

ACCESSORIES

-608

-12880

1

3

143

0.00

2005-05-21 15:34:16

0.281

1.586

17.463

Y

7.301

210

52.55793

52.57100

-169.7829

-169.7910

3-6

212

219

NA

6.1

NA

549

172

129

-608

-12881

2

3

143

1.11

2005-05-21 18:51:09

0.252

1.331

15.101

Y

7.292

10

52.63840

52.62704

-169.7815

-169.7778

3-8

75

82

6

5.3

4.9

320

172

129

-608

-12882

3

3

143

0.00

2005-05-22 13:19:04

0.255

1.423

16.943

Y

6.808

111

52.67131

52.68121

-169.4279

-169.4403

7-9

176

183

NA

5.3

4.7

503

172

129

### 8.2.6 AKFIN\_LENGTH

Number of rows: 2449719

Number of columns: 7

HAULJOIN

SPECIES\_CODE

SEX

FREQUENCY

LENGTH

LENGTH\_TYPE

SAMPLE\_TYPE

-12880

10110

2

1

160

1

1

-12880

10110

2

1

210

1

1

-12880

10110

1

1

250

1

1

### 8.2.7 AKFIN\_METADATA\_COLUMN

Number of rows: 133

Number of columns: 5

METADATA\_COLNAME

METADATA\_COLNAME\_LONG

METADATA\_UNITS

METADATA\_DATATYPE

METADATA\_COLNAME\_DESC

VESSEL\_COAST\_GUARD\_NUMBER

Vessel Coast Guard Number

ID code

NUMBER(38,0)

Official Identification number as defined by www.dco.uscg.mil. The Official Number (O/N) is the 6 or 7 digit number awarded to the vessel at the time it is first documented with the US Coast Guard. This number remains with the vessel indefinitely and should be marked in accordance with 46 CFR 67.121.

VESSEL\_ID

Vessel ID

ID code

NUMBER(38,0)

ID number of the vessel used to collect data for that haul. The column ‘vessel\_id’ is associated with the ‘vessel\_name’ column. Note that it is possible for a vessel to have a new name but the same vessel id number. For a complete list of vessel ID codes, review the [code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual).

VESSEL\_IMO

Vessel International Maritime Organization Number

ID code

NUMBER(38,0)

The International Maritime Organization (IMO) number consists of the letters "IMO" followed by a unique, seven-digit number: the pattern is "NNNNNNN", where N is a single-digit number, e.g., "1234567"

### 8.2.8 AKFIN\_SIZECOMP

Number of rows: 3439200

Number of columns: 7

SURVEY\_DEFINITION\_ID

YEAR

AREA\_ID

SPECIES\_CODE

LENGTH\_MM

SEX

POPULATION\_COUNT

143

2021

99902

10110

540

2

43378

143

2021

99902

10110

560

2

70492

143

2021

99902

10110

570

2

43378

### 8.2.9 AKFIN\_SPECIMEN

Number of rows: 343378

Number of columns: 17

CRUISEJOIN

HAULJOIN

REGION

VESSEL\_ID

SPECIMEN\_ID

STRATUM

SPECIES\_CODE

LENGTH\_MM

SEX

WEIGHT\_G

AGE\_YEARS

MATURITY

MATURITY\_TABLE

GONAD\_G

SPECIMEN\_SUBSAMPLE\_METHOD

SPECIMEN\_SAMPLE\_TYPE

AGE\_DETERMINATION\_METHOD

-608

-12880

GOA

143

1

0

21921

400

1

792

6

NA

NA

NA

5

1

4

-608

-12880

GOA

143

2

0

21921

380

1

810

6

NA

NA

NA

5

1

4

-608

-12880

GOA

143

3

0

21921

390

1

822

7

NA

NA

NA

5

1

4

### 8.2.10 AKFIN\_STRATUM\_GROUPS

Number of rows: 744

Number of columns: 4

AREA\_ID

SURVEY\_DEFINITION\_ID

DESIGN\_YEAR

STRATUM

799

52

1980

721

799

52

1980

712

799

52

1980

722

### 8.2.11 AKFIN\_SURVEY\_DESIGN

Number of rows: 126

Number of columns: 3

SURVEY\_DEFINITION\_ID

YEAR

DESIGN\_YEAR

52

1980

1980

52

1983

1980

52

1986

1980

### 8.2.12 AKFIN\_CRUISE

[There is currently no description for this table.]

Number of rows: 188

Number of columns: 10

CRUISEJOIN

CRUISE

YEAR

SURVEY\_DEFINITION\_ID

SURVEY\_NAME

VESSEL\_ID

VESSEL\_NAME

SPONSOR\_ACRONYM

DATE\_START

DATE\_END

-717

201702

2017

143

Northern Bering Sea Crab/Groundfish Survey - Eastern Bering Sea Shelf Survey Extension

94

VESTERAALEN

NMFS/S

2017-08-08

2017-08-30

-718

201702

2017

143

Northern Bering Sea Crab/Groundfish Survey - Eastern Bering Sea Shelf Survey Extension

162

ALASKA KNIGHT

NMFS/S

2017-08-08

2017-08-30

-660

201002

2010

143

Northern Bering Sea Crab/Groundfish Survey - Eastern Bering Sea Shelf Survey Extension

162

ALASKA KNIGHT

NWAFC/S

2010-08-04

2010-08-11

### 8.2.13 AKFIN\_TAXONOMICS\_WORMS

All cruise, vessel, and timing data for all relevant, standard cruises. These data were calculated using all standard hauls that can be applied to production data and abundance calculations. These data are produced by by the Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC). There are legal restrictions on access to the data. These data are not intended for public dissemination and should not be shared without the explicit written consent of the data managers and owners (NOAA Fisheries). The data from this dataset are shared on the Fisheries One Stop Stop (FOSS) platform (https://www.fisheries.noaa.gov/foss/). The GitHub repository for the scripts that created this code can be found at https://github.com/afsc-gap-products/gap\_products. For more information about codes used in the tables, please refer to the survey code books (https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual). These data were last updated June 27, 2023.

Number of rows: 2762

Number of columns: 21

REASON

SURVEY\_NAME

ACCEPTED\_NAME

COMMON\_NAME

SPECIES\_CODE

ID\_RANK

DATABASE

DATABASE\_ID

GENUS

SUBFAMILY

FAMILY

SUPERFAMILY

SUBORDER

ORDER

SUPERORDER

SUBCLASS

CLASS

SUPERCLASS

SUBPHYLUM

PHYLUM

KINGDOM

NA

Anarrhichthys ocellatus

Anarrhichthys ocellatus

wolf-eel

20320

species

WORMS

279605

Anarrhichthys

NA

Anarhichadidae

NA

Zoarcoidei

Perciformes

NA

NA

Teleostei

Actinopteri

Vertebrata

Chordata

Animalia

NA

Anatoma crispata

Anatoma crispata

crispate scissurelle

71333

species

WORMS

141417

Anatoma

NA

Anatomidae

Scissurelloidea

NA

Lepetellida

NA

Vetigastropoda

Gastropoda

NA

NA

Mollusca

Animalia

synon

Thieleella baxteri

Anatoma lyra

NA

71332

species

WORMS

492991

Anatoma

NA

Anatomidae

Scissurelloidea

NA

Lepetellida

NA

Vetigastropoda

Gastropoda

NA

NA

Mollusca

Animalia

# 9. AFSC RACE Groundfish Data for AKFIN in Oracle with SQL and R

## 9.1 Access data via Oracle (AFSC only)

AFSC Oracle users can access the database via SQL developer to view and pull the production data directly from the GAP\_PRODUCTS Oracle schema.

### 9.1.1 Access data via Oracle

If the user has access to the AFSC Oracle database, the user can use SQL developer to view and pull the GAP Products data directly from the GAP\_PRODUCTS Oracle schema.

### 9.1.2 Connect to Oracle from R

Many users will want to access the data from Oracle using R. The user will need to install the RODBC R package and ask OFIS (IT) connect R to Oracle. Then, use the following code in R to establish a connection from R to Oracle:

Here, the user can establish the oracle connection by entering their username and password in the channel <- gapindex::oracle\_connect() function. Never save usernames or passwords in scripts that may be intentionally or unintentionally shared with others. If no username and password is entered in the function, pop-ups will appear on the screen asking for the username and password.

## 9.2 Data SQL Query Examples:

### 9.2.1 Ex. 0: Select all data from a table

You can download all of the tables locally using a variation of the code below. Once connected, pull and save the tables of interest into the R environment.

locations <- c(  
 "GAP\_PRODUCTS.AKFIN\_AGECOMP",   
 "GAP\_PRODUCTS.AKFIN\_AREA",   
 "GAP\_PRODUCTS.AKFIN\_BIOMASS",   
 "GAP\_PRODUCTS.AKFIN\_CATCH",   
 "GAP\_PRODUCTS.AKFIN\_CPUE",   
 "GAP\_PRODUCTS.AKFIN\_CRUISE",   
 "GAP\_PRODUCTS.AKFIN\_HAUL",   
 "GAP\_PRODUCTS.AKFIN\_LENGTH",   
 "GAP\_PRODUCTS.AKFIN\_METADATA\_COLUMN",   
 "GAP\_PRODUCTS.AKFIN\_SIZECOMP",   
 "GAP\_PRODUCTS.AKFIN\_SPECIMEN",   
 "GAP\_PRODUCTS.AKFIN\_STRATUM\_GROUPS",   
 "GAP\_PRODUCTS.AKFIN\_SURVEY\_DESIGN",   
 "GAP\_PRODUCTS.AKFIN\_TAXONOMICS\_WORMS"   
)  
  
for (i in 1:length(locations)) {  
 print(locations[i])  
 a <- RODBC::sqlQuery(channel, paste0("SELECT \* FROM ", locations[i]))  
 write.csv(x = a, file = here::here("data", paste0(locations[i], ".csv")))  
}

### 9.2.2 Ex. 1: GOA Pacific Ocean perch biomass and abundance

Biomass and abundance for Pacific Ocean perch from 1990 – 2023 for the western/central/eastern GOA management areas as well as for the entire region.

dat <- RODBC::sqlQuery(channel = channel,   
 query =   
"WITH FILTERED\_STRATA AS (  
SELECT AREA\_ID, DESCRIPTION FROM GAP\_PRODUCTS.AKFIN\_AREA  
WHERE TYPE in ('REGULATORY\_AREA', 'REGION')   
AND SURVEY\_DEFINITION\_ID = 47)  
SELECT   
BIOMASS\_MT,  
POPULATION\_COUNT,   
YEAR,   
DESCRIPTION  
FROM GAP\_PRODUCTS.AKFIN\_BIOMASS BIOMASS  
JOIN FILTERED\_STRATA STRATA   
ON STRATA.AREA\_ID = BIOMASS.AREA\_ID  
WHERE BIOMASS.SURVEY\_DEFINITION\_ID IN 47   
AND BIOMASS.SPECIES\_CODE = 30060")

dat0 <- dat %>%   
 janitor::clean\_names() %>%   
 dplyr::select(biomass\_mt, population\_count, year, area = description) %>%  
 pivot\_longer(cols = c("biomass\_mt", "population\_count"),   
 names\_to = "var",   
 values\_to = "val") %>%   
 dplyr::mutate(  
 val = ifelse(var == "biomass\_mt", val/1e6, val/1e9),   
 var = ifelse(var == "biomass\_mt", "Biomass (Mmt)", "Population (B)"),   
 area = gsub(x = area, pattern = " - ", replacement = "\n"),   
 area = gsub(x = area, pattern = ": ", replacement = "\n"),   
 type = sapply(X = strsplit(x = area, split = "\n", fixed = TRUE), `[[`, 2)) %>%   
 dplyr::arrange(type) %>%   
 dplyr::mutate(  
 area = factor(area, levels = unique(area), labels = unique(area), ordered = TRUE))  
  
flextable::flextable(head(dat)) %>% theme\_zebra()

| **BIOMASS\_MT** | **POPULATION\_COUNT** | **YEAR** | **DESCRIPTION** |
| --- | --- | --- | --- |
| 483,622.6 | 833,902,161 | 1,993 | GOA Region: All Strata |
| 771,412.8 | 1,252,616,603 | 1,996 | GOA Region: All Strata |
| 727,063.5 | 1,212,034,913 | 1,999 | GOA Region: All Strata |
| 673,155.1 | 1,189,370,120 | 2,001 | GOA Region: All Strata |
| 457,421.6 | 781,034,228 | 2,003 | GOA Region: All Strata |
| 764,901.4 | 1,343,536,275 | 2,005 | GOA Region: All Strata |

# install.packages("scales")  
library(scales)  
figure <- ggplot2::ggplot(  
 dat = dat0,   
 mapping = aes(x = year, y = val, color = type)) +  
 ggplot2::geom\_point(size = 3) +   
 ggplot2::facet\_grid(cols = vars(area), rows = vars(var), scales = "free\_y") +   
 ggplot2::scale\_x\_continuous(name = "Year", n.breaks = 3) +  
 ggplot2::scale\_y\_continuous(name = "Estimate", labels = comma) +  
 ggplot2::labs(title = 'GOA Pacific Ocean perch biomass and abundance 1990 – 2023') +   
 ggplot2::guides(color=guide\_legend(title = "Region Type"))+  
 ggplot2::scale\_color\_grey() +  
 ggplot2::theme\_bw() +  
 ggplot2::theme(legend.direction = "horizontal",   
 legend.position = "bottom")  
  
figure

|  |
| --- |
| Ex. 1: GOA Pacific Ocean perch biomass and abundance. |

### 9.2.3 Ex. 2: AI Rock sole size compositions and ridge plot

Northern and Southern rock sole size composition data from 1991 – 2022 for the Aleutian Islands, with Ridge plot from [ggridges](https://cran.r-project.org/web/packages/ggridges/vignettes/introduction.html).

dat <- RODBC::sqlQuery(channel = channel,   
 query =   
"WITH FILTERED\_STRATA AS (  
SELECT   
AREA\_ID,   
DESCRIPTION   
FROM GAP\_PRODUCTS.AKFIN\_AREA  
WHERE TYPE = 'REGION'   
AND SURVEY\_DEFINITION\_ID = 52)  
SELECT   
LENGTH\_MM,   
YEAR  
FROM GAP\_PRODUCTS.AKFIN\_SIZECOMP SIZECOMP  
JOIN FILTERED\_STRATA STRATA   
ON STRATA.AREA\_ID = SIZECOMP.AREA\_ID  
WHERE SIZECOMP.SURVEY\_DEFINITION\_ID IN 52   
AND SIZECOMP.SPECIES\_CODE IN (10261, 10262)")

dat0 <- dat %>%   
 janitor::clean\_names() %>%   
 dplyr::mutate(length\_cm = length\_mm/10)  
flextable::flextable(head(dat)) %>% theme\_zebra()

| **LENGTH\_MM** | **YEAR** |
| --- | --- |
| 110 | 1,997 |
| 130 | 1,997 |
| 140 | 1,997 |
| 150 | 1,997 |
| 160 | 1,997 |
| 170 | 1,997 |

# install.packages("ggridges")  
library(ggridges)  
figure <-   
 ggplot2::ggplot(  
 data = dat0,   
 mapping = aes(x = length\_cm, y = as.factor(year), fill = stat(x))) +  
 ggridges::theme\_ridges(center\_axis\_labels = TRUE) +   
 ggridges::geom\_density\_ridges\_gradient(scale = 4, show.legend = FALSE) +   
 ggplot2::scale\_y\_discrete(name = "Year", expand = c(0.01, 0)) +  
 ggplot2::scale\_x\_continuous(name = "Length (cm)", expand = c(0.01, 0)) +  
 # ggplot2::scale\_fill\_grey() +  
 ggplot2::labs(title = 'AI Rock sole Size Compositions 1991 – 2022')   
  
figure

|  |
| --- |
| Ex. 2: AI Rock sole size compositions and ridge plot. |

### 9.2.4 Ex. 3: EBS Walleye Pollock Age Compositions and Age Pyramid

Walleye pollock age composition for the EBS Standard Area from 1982 – 2022 and the EBS + NW Area from 1987 – 2022, with age pyramid plot.

dat <- RODBC::sqlQuery(channel = channel,   
 query =   
"WITH FILTERED\_STRATA AS (  
SELECT   
AREA\_ID,   
DESCRIPTION   
FROM GAP\_PRODUCTS.AKFIN\_AREA  
WHERE TYPE = 'REGION' AND   
SURVEY\_DEFINITION\_ID = 98)  
SELECT   
AGECOMP.AGE,   
AGECOMP.POPULATION\_COUNT,   
AGECOMP.SEX  
FROM GAP\_PRODUCTS.AKFIN\_AGECOMP AGECOMP  
JOIN FILTERED\_STRATA STRATA   
ON STRATA.AREA\_ID = AGECOMP.AREA\_ID  
WHERE SURVEY\_DEFINITION\_ID = 98   
AND SPECIES\_CODE = 21740  
AND AGE >= 0")

dat0 <- dat %>%   
 janitor::clean\_names() %>%   
 dplyr::filter(sex %in% c(1,2)) %>%  
 dplyr::mutate(  
 sex = ifelse(sex == 1, "M", "F"),  
 population\_count = # change male population to negative  
 ifelse(sex=="M", population\_count\*(-1), population\_count\*1)/1e9)   
  
flextable::flextable(head(dat)) %>% theme\_zebra()

| **AGE** | **POPULATION\_COUNT** | **SEX** |
| --- | --- | --- |
| 1 | 33,930,956 | 1 |
| 2 | 314,043,443 | 1 |
| 3 | 103,452,658 | 1 |
| 4 | 47,525,134 | 1 |
| 5 | 203,340,101 | 1 |
| 6 | 246,665,076 | 1 |

figure <- ggplot2::ggplot(  
 data = dat0,   
 mapping =   
 aes(x = age,  
 y = population\_count,   
 fill = sex)) +  
 ggplot2::scale\_fill\_grey() +  
 ggplot2::geom\_bar(stat = "identity") +  
 ggplot2::coord\_flip() +  
 ggplot2::scale\_x\_continuous(name = "Age") +  
 ggplot2::scale\_y\_continuous(name = "Population (billions)", labels = abs) +  
 ggplot2::ggtitle(label = "EBS Walleye Pollock Age Compositions 1982 – 2022") +   
 ggplot2::guides(fill = guide\_legend(title = "Sex"))+  
 ggplot2::theme\_bw()  
  
figure

|  |
| --- |
| Ex. 3: EBS Walleye Pollock Age Compositions and Age Pyramid. |

### 9.2.5 Ex. 4: NBS Pacific cod biomass and abundance

Pacific cod biomass and abundance data for the NBS by stratum.

dat <- RODBC::sqlQuery(channel = channel,   
 query =   
"WITH FILTERED\_STRATA AS (  
SELECT   
AREA\_ID,   
AREA\_NAME,   
DESCRIPTION   
FROM GAP\_PRODUCTS.AKFIN\_AREA  
WHERE TYPE in ('STRATUM') AND   
SURVEY\_DEFINITION\_ID = 143)   
SELECT   
BIOMASS.BIOMASS\_MT,   
BIOMASS.POPULATION\_COUNT,   
BIOMASS.YEAR,   
STRATA.AREA\_NAME  
FROM GAP\_PRODUCTS.AKFIN\_BIOMASS BIOMASS   
JOIN FILTERED\_STRATA STRATA   
ON STRATA.AREA\_ID = BIOMASS.AREA\_ID  
WHERE BIOMASS.SURVEY\_DEFINITION\_ID IN 143   
AND BIOMASS.SPECIES\_CODE = 21720")

dat0 <- dat %>%   
 janitor::clean\_names() %>%   
 dplyr::select(biomass\_mt, population\_count, year, area = area\_name) %>%  
 pivot\_longer(cols = c("biomass\_mt", "population\_count"),   
 names\_to = "var",   
 values\_to = "val") %>%   
 dplyr::mutate(  
 val = ifelse(var == "biomass\_mt", val/1e6, val/1e9),   
 var = ifelse(var == "biomass\_mt", "Biomass (Mmt)", "Population (B)"),   
 area = factor(area, levels = unique(area), labels = unique(area), ordered = TRUE))  
flextable::flextable(head(dat)) %>% theme\_zebra()

| **BIOMASS\_MT** | **POPULATION\_COUNT** | **YEAR** | **AREA\_NAME** |
| --- | --- | --- | --- |
| 7,462.559 | 4,724,153 | 2,010 | Inner Domain |
| 95,849.983 | 68,767,498 | 2,021 | Inner Domain |
| 107,096.730 | 102,734,142 | 2,019 | Inner Domain |
| 132,490.152 | 66,187,245 | 2,017 | Inner Domain |
| 96,500.697 | 60,433,135 | 2,022 | Inner Domain |
| 147,971.454 | 65,078,489 | 2,017 | Inner Domain |

figure <- ggplot2::ggplot(  
 dat = dat0,   
 mapping = aes(y = val, x = year, fill = area)) +   
 ggplot2::geom\_bar(position="stack", stat="identity") +   
 ggplot2::facet\_grid(rows = vars(var), scales = "free\_y") +  
 ggplot2::scale\_y\_continuous(name = "Estimate", labels = comma) +  
 ggplot2::scale\_x\_continuous(name = "Year", breaks = unique(dat0$year)) +  
 ggplot2::labs(title = 'NBS Pacific cod biomass and abundance by stratum') +   
 ggplot2::guides(fill=guide\_legend(title = "Region Type"))+  
 ggplot2::scale\_fill\_grey() +  
 ggplot2::theme\_bw() +  
 ggplot2::theme(legend.direction = "horizontal",   
 legend.position = "bottom")  
  
figure

|  |
| --- |
| Ex. 4: NBS Pacific cod biomass and abundance. |

### 9.2.6 Ex. 5: GOA Pacific Ocean perch biomass and line plot

Pacific Ocean perch biomass totals for GOA between 1984-2021 from GAP\_PRODUCTS.AKFIN\_BIOMASS

dat <- RODBC::sqlQuery(channel = channel,   
 query =   
"SELECT   
SURVEY\_DEFINITION\_ID,   
BIOMASS\_MT,   
YEAR  
FROM GAP\_PRODUCTS.AKFIN\_BIOMASS  
WHERE SPECIES\_CODE = 30060   
AND SURVEY\_DEFINITION\_ID = 47   
AND AREA\_ID = 99903   
AND YEAR BETWEEN 1984 AND 2021;") %>%   
 janitor::clean\_names() %>%   
 dplyr::mutate(biomass\_mt = biomass\_mt/1000)

a\_mean <- dat %>%   
 dplyr::group\_by(survey\_definition\_id) %>%   
 dplyr::summarise(biomass\_mt = mean(biomass\_mt, na.rm = TRUE),   
 minyr = min(year, na.rm = TRUE),   
 maxyr = max(year, na.rm = TRUE))   
flextable::flextable(head(dat)) %>% theme\_zebra()

| **survey\_definition\_id** | **biomass\_mt** | **year** |
| --- | --- | --- |
| 47 | 483.6226 | 1,993 |
| 47 | 771.4128 | 1,996 |
| 47 | 727.0635 | 1,999 |
| 47 | 673.1551 | 2,001 |
| 47 | 457.4216 | 2,003 |
| 47 | 764.9014 | 2,005 |

figure <-  
 ggplot(data = dat,   
 mapping = aes(x = year,   
 y = biomass\_mt)) +  
 ggplot2::geom\_point(size = 2.5, color = "grey40") +   
 ggplot2::scale\_x\_continuous(  
 name = "Year",   
 labels = scales::label\_number(  
 accuracy = 1,   
 big.mark = "")) +  
 ggplot2::scale\_y\_continuous(  
 name = "Biomass (Kmt)",   
 labels = comma) +  
 ggplot2::geom\_segment(  
 data = a\_mean,  
 mapping = aes(x = minyr,   
 xend = maxyr,   
 y = biomass\_mt,   
 yend = biomass\_mt),  
 linetype = "dashed",   
 linewidth = 2) +  
 ggplot2::ggtitle(  
 label = "GOA Pacific Ocean Perch Biomass 1984-2021",   
 subtitle = paste0("Mean = ",   
 formatC(x = a\_mean$biomass\_mt,   
 digits = 2,   
 big.mark = ",",   
 format = "f"),   
 " Kmt")) +  
 ggplot2::theme\_bw()  
  
figure

|  |
| --- |
| Ex. 5: GOA Pacific Ocean perch biomass and line plot. |

### 9.2.7 Ex. 6: EBS Pacific Ocean perch CPUE and [akgfmaps](https://github.com/afsc-gap-products/akgfmaps) map

Pacific Ocean perch catch-per-unit-effort estimates for EBS in 2021 from GAP\_PRODUCTS.AKFIN\_CPUE and map constructed using [akgfmaps](https://github.com/afsc-gap-products/akgfmaps). Here, we’ll use AKFIN HAUL and CRUISES data also included in this repo, for convenience, though they are very similar to their RACEBASE analogs.

dat <- RODBC::sqlQuery(channel = channel,   
 query =   
"SELECT   
(cp.CPUE\_KGKM2/100) CPUE\_KGHA, -- akgfmaps is expecting hectares  
hh.LATITUDE\_DD\_START LATITUDE,  
hh.LONGITUDE\_DD\_START LONGITUDE  
  
FROM GAP\_PRODUCTS.AKFIN\_CPUE cp  
  
-- Use HAUL data to obtain LATITUDE & LONGITUDE and connect to cruisejoin  
LEFT JOIN GAP\_PRODUCTS.AKFIN\_HAUL hh  
ON cp.HAULJOIN = hh.HAULJOIN  
  
-- Use CRUISES data to obtain YEAR and SURVEY\_DEFINITION\_ID  
LEFT JOIN GAP\_PRODUCTS.AKFIN\_CRUISE cc  
ON hh.CRUISEJOIN = cc.CRUISEJOIN  
  
WHERE cp.SPECIES\_CODE = 30060   
AND cc.SURVEY\_DEFINITION\_ID = 98   
AND cc.YEAR = 2021;")

flextable::flextable(head(dat)) %>% theme\_zebra()

| **CPUE\_KGHA** | **LATITUDE** | **LONGITUDE** |
| --- | --- | --- |
| 0.00000000 | 58.66802 | -176.1673 |
| 0.00000000 | 60.69381 | -175.4619 |
| 0.00000000 | 58.97738 | -173.0898 |
| 0.00000000 | 61.68338 | -173.6652 |
| 0.00000000 | 60.65295 | -176.2033 |
| 0.03091028 | 59.97384 | -176.7033 |

# devtools::install\_github("afsc-gap-products/akgfmaps", build\_vignettes = TRUE)  
library(akgfmaps)  
  
figure <- akgfmaps::make\_idw\_map(  
 x = dat, # Pass data as a data frame  
 region = "bs.south", # Predefined EBS area  
 set.breaks = "jenks", # Gets Jenks breaks from classint::classIntervals()  
 in.crs = "+proj=longlat", # Set input coordinate reference system  
 out.crs = "EPSG:3338", # Set output coordinate reference system  
 grid.cell = c(20000, 20000), # 20x20km grid  
 key.title = "Pacific Ocean perch") # Include in the legend title

[inverse distance weighted interpolation]  
[inverse distance weighted interpolation]

figure$plot +   
 ggplot2::guides(fill=guide\_legend(title = "Pacific Ocean perch\nCPUE (kg/km2)")) |>   
 change\_fill\_color(new.scheme = "grey", show.plot = FALSE)

|  |
| --- |
| Ex. 6: EBS Pacific Ocean perch CPUE and [akgfmaps](https://github.com/afsc-gap-products/akgfmaps) map. |

# 10. GAP FOSS Column Metadata

## 10.1 Data description

The Resource Assessment and Conservation Engineering Division (RACE) Groundfish Assessment Program (GAP) of the Alaska Fisheries Science Center (AFSC) conducts fisheries-independent bottom trawl surveys to monitor the condition of the demersal fish and crab stocks of Alaska. These data are developed to describe the temporal distribution and abundance of commercially and ecologically important groundfish species, examine the changes in the species composition of the fauna over time and space, and describe the physical environment of the groundfish habitat.

There are no legal restrictions on access to the data. They reside in the public domain and can be freely distributed. Users must read and fully comprehend the metadata prior to use. Data should not be used beyond the limits of the source scale. Acknowledgement of NOAA, as the source from which these data were obtained, in any publications and/or other representations of these data, is suggested. These data are compiled and approved annually after each summer survey season. The data from previous years are unlikely to change substantially once published.

These data are zero-filled (presence and absence) observations from surveys conducted on fishing vessels. These surveys monitor trends in distribution and abundance of groundfish, crab, and bottom-dwelling species in Alaska’s marine ecosystems. These data include estimates of catch-per-unit-effort (CPUE) for all identified species for index stations. Some survey data are excluded, such as non-standard stations, surveys completed in earlier years using different/non-standard gear, and special tows and non-standard data collections.

Though not included in the public data, these surveys also collect oceanographic and environmental data, and biological data such as length, weight, stomach contents (to learn more about diet), otoliths (fish ear bones to learn about age), and tissue samples for genetic analysis, all of which can be shared upon special request. Also not included in the public data are estimated biomass (average total weight of all fish and crabs sampled) of crabs and groundfish that support the creation of annual stock assessments.

## 10.2 Data created in this repo

### 10.2.1 FOSS\_HAUL

Number of rows: 32510

Number of columns: 27

YEAR

SRVY

SURVEY

SURVEY\_DEFINITION\_ID

SURVEY\_NAME

CRUISE

CRUISEJOIN

HAULJOIN

HAUL

STRATUM

STATION

VESSEL\_ID

VESSEL\_NAME

DATE\_TIME

LATITUDE\_DD\_START

LONGITUDE\_DD\_START

LATITUDE\_DD\_END

LONGITUDE\_DD\_END

BOTTOM\_TEMPERATURE\_C

SURFACE\_TEMPERATURE\_C

DEPTH\_M

DISTANCE\_FISHED\_KM

DURATION\_HR

NET\_WIDTH\_M

NET\_HEIGHT\_M

AREA\_SWEPT\_KM2

PERFORMANCE

2022

AI

Aleutian Islands

52

Aleutian Islands Bottom Trawl Survey

202201

-753

-21932

220

321

111-21

176

ALASKA PROVIDER

2022-08-10 12:45:10

51.86266

178.2546

51.85674

178.2753

5.2

6.5

91

1.584

0.279

19.314

6.033

0.030593

0

2022

AI

Aleutian Islands

52

Aleutian Islands Bottom Trawl Survey

202201

-754

-21872

208

223

86-32

148

OCEAN EXPLORER

2022-08-05 09:35:49

52.34030

176.3832

52.35010

176.3747

4.6

10.2

232

1.238

0.223

17.174

5.732

0.021261

0

2022

AI

Aleutian Islands

52

Aleutian Islands Bottom Trawl Survey

202201

-754

-21893

217

222

79-24

148

OCEAN EXPLORER

2022-08-07 12:24:07

52.01721

175.8889

52.00690

175.9017

4.6

9.7

184

1.449

0.263

16.387

6.345

0.023745

0

### 10.2.2 FOSS\_CATCH

[There is currently no description for this table.]

Number of rows: 42281918

Number of columns: 12

HAULJOIN

SPECIES\_CODE

CPUE\_KGKM2

CPUE\_NOKM2

COUNT

WEIGHT\_KG

TAXON\_CONFIDENCE

SCIENTIFIC\_NAME

COMMON\_NAME

ID\_RANK

WORMS

ITIS

-21974

44102

0

0

0

0

NA

Plumarella echinata

NA

species

574239

NA

-21974

79000

0

0

0

0

NA

Decapodiformes

squid unid.

NA

325342

NA

-21974

80660

0

0

0

0

NA

Pseudarchaster parelii

scarlet sea star

species

124085

157017

### 10.2.3 FOSS\_CPUE\_PRESONLY

[There is currently no description for this table.]

Number of rows: 42281918

Number of columns: 37

YEAR

SRVY

SURVEY

SURVEY\_DEFINITION\_ID

CRUISE

CRUISEJOIN

HAUL

HAULJOIN

STRATUM

STATION

VESSEL\_ID

VESSEL\_NAME

DATE\_TIME

LATITUDE\_DD\_START

LONGITUDE\_DD\_START

LATITUDE\_DD\_END

LONGITUDE\_DD\_END

BOTTOM\_TEMPERATURE\_C

SURFACE\_TEMPERATURE\_C

DEPTH\_M

SPECIES\_CODE

ITIS

WORMS

COMMON\_NAME

SCIENTIFIC\_NAME

ID\_RANK

TAXON\_CONFIDENCE

WEIGHT\_KG

COUNT

CPUE\_KGKM2

CPUE\_NOKM2

AREA\_SWEPT\_KM2

DISTANCE\_FISHED\_KM

DURATION\_HR

NET\_WIDTH\_M

NET\_HEIGHT\_M

PERFORMANCE

2022

NBS

northern Bering Sea

143

202202

-758

75

-22026

70

R-02

162

ALASKA KNIGHT

2022-08-20 13:41:41

60.67853

-167.3545

60.65290

-167.3347

9.9

9.9

24

23843

171570

254578

bearded warbonnet

Chirolophis snyderi

species

3

0

0

0

0

0.044556

3.047

0.534

14.623

2.315

0

2022

NBS

northern Bering Sea

143

202202

-758

74

-22025

70

R-01

162

ALASKA KNIGHT

2022-08-20 10:47:02

60.68382

-168.0264

60.65717

-168.0218

9.4

9.5

28

71713

73953

423285

NA

Colus capponius

species

NA

0

0

0

0

0.044405

2.972

0.527

14.941

2.257

0

2022

NBS

northern Bering Sea

143

202202

-758

72

-22023

70

R-20

162

ALASKA KNIGHT

2022-08-19 16:08:03

60.66940

-170.0343

60.67139

-170.0865

4.3

7.4

50

99998

NA

NA

NA

Polychaeta tubes

class

1

0

0

0

0

0.047321

2.869

0.519

16.494

1.975

0

### 10.2.4 FOSS\_TAXON\_GROUPS

[There is currently no description for this table.]

Number of rows: 15775

Number of columns: 3

ID\_RANK

CLASSIFICATION

SPECIES\_CODE

class

Asteroidea

80525

class

Asteroidea

80535

class

Asteroidea

80536

# 11. Access public data using the FOSS Interactive Platform

|  |
| --- |
| AFSC Groundfish and Crab Assessment Program Bottom Trawl Survey data interface on the Fisheries One Stop Shop platform. |

## 11.1 Select and filter

Select, filter, and package this and other NOAA Fisheries data from the [Fisheries One Stop Shop (FOSS)](https://www.fisheries.noaa.gov/foss) platform. A user guide for the FOSS platform can be found [here](https://www.fisheries.noaa.gov/foss/f?p=215:7:7542600605674:::::). To begin a report, select options from the boxes what you need data for.

For a given box, select one or a few options from the “options box” (list on the left) to query by highlighting them. To select multiple options, hold down the CTRL key while clicking on the options of interest, or click and drag down the list. Once the options you wish to be included in your query are highlighted, click the right-pointing arrow (>) to move them into the “selection box” (list on the right). If you accidentally select an option that you do not want to query, simply select the unwanted option from the selection box and click the left-pointing arrow (<).

If you wish to select all options from the options box and send them to the selection box, simply click the double right-pointing arrow (>>). If you want to unselect all options from the selection box, use the double left-pointing arrow (<<) or the reset icon.

To find a specific species or group more quickly you can use the Search Species option to quickly narrow the options. Search for parts of species common names in the Search Species box by entering a term and clicking the search button. The platform will return a shorter list in the Speices options box of only species that contain a match to that search term.

Use the Reset All Parameters button to reset all parameters for entire form.

|  |
| --- |
| Diagram of selection and search tools available on the FOSS platofrom. |

Filter options:

* Survey: Each survey has different in design, time series, and history. More information on each survey and their designs can be found in our [annual data reports](https://www.fisheries.noaa.gov/alaska/science-data/groundfish-assessment-program-bottom-trawl-surveys#data-products).
* Year: Surveys are not conducted in all years, so only data from the years for which the survey was conducted will be returned.
* Species: Common name of all species ever encountered in the survey. Find more information about these species in our [survey code books](https://www.fisheries.noaa.gov/resource/document/groundfish-survey-species-code-manual-and-data-codes-manual).

In this example, we’ll select for 2022 eastern Bering Sea Pacific cod data. Here, we used the Search Species box to search for species with the term “cod” in their common names and selected “Pacific cod” from that shortened list.

|  |
| --- |
| Diagram of selection and search tools available on the FOSS platofrom. |

## 11.2 Select data format

Select from the below radio list of pre-designed output tables. Once you run the report, the user can further specify filter data and select columns of interest. The tables below will only include data from the selections made in the previous step.

* All Data Fields: Presence and Absence (zero-filled): The most complete version of the data, including species, catch, haul, and environmental data. This data will include catch data for where species were caught and zeros for where the species were not caught. This is important for calculating catch-per-unit-effort data, preparing distribution plots (e.g., [using the akgfmaps R package](https://github.com/afsc-gap-products/akgfmaps)), and many statistical analyses.
* All Data Fields: Presence-only (non-zero): The second most complete version of the data, including species, catch, haul, and environmental data. However, this data only includes catch data for where species were caught and does not include zeros for where the species were not caught. This will return smaller, more focused data and can be useful for quickly assessing how many species were caught or how many stations species were caught at.
* Catch data: Presence and Absence (zero-filled): This data set is similar to All Data Fields: Presence and Absence (zero-filled), but only includes catch and species data columns.
* Catch data: Presence-only (non-zero): This data set is similar to All Data Fields: Presence-only (non-zero), but only includes catch and species data columns.
* Haul Data: This data set only includes haul and environmental data collected from the survey. This data will only include one observation per haul event/station.

In this example, we’ll select All Data Fields: Presence and Absence (zero-filled).

|  |
| --- |
| Diagram of the pre-set data format options. |

## 11.3 Run report

Click the RUN REPORT button. Below the select and filter area, the results of your query will appear below the page in the format you selected. To change the format, make a different selection and run the report again. Further modifications to your results can be made by clicking on the Actions button above your data. Here you can download your data, select columns included in your results, and apply a variety of filters and mathematical tools.

|  |
| --- |
| Example data returned from running the report. |

# 12. Access public data through the FOSS API using R

An application programming interface (API) is a way for two or more computer programs to communicate with each other.

More information about how to amend API links can be found [here](https://docs.oracle.com/en/database/oracle/oracle-rest-data-services/22.3/books.html#AELIG90103/). Useful introductions to using APIs in R can be found [here](https://www.dataquest.io/blog/r-api-tutorial/).

## 12.1 Ex. 1: Load the first 25 rows (default) of data

# install.packages(c("httr", "jsonlite"))  
library(httr)  
library(jsonlite)  
library(dplyr)  
 # link to the API  
api\_link <- "https://apps-st.fisheries.noaa.gov/ods/foss/afsc\_groundfish\_survey/"  
  
res <- httr::GET(url = api\_link)  
 # res # Test connection  
data <- jsonlite::fromJSON(base::rawToChar(res$content))  
 # names(data)  
flextable::flextable(head(data$items, 3))

| year | srvy | survey | survey\_id | cruise | haul | stratum | station | vessel\_name | vessel\_id | date\_time | latitude\_dd | longitude\_dd | species\_code | common\_name | scientific\_name | taxon\_confidence | cpue\_kgha | cpue\_kgkm2 | cpue\_kg1000km2 | cpue\_noha | cpue\_nokm2 | cpue\_no1000km2 | weight\_kg | count | bottom\_temperature\_c | surface\_temperature\_c | depth\_m | distance\_fished\_km | net\_width\_m | net\_height\_m | area\_swept\_ha | duration\_hr | tsn | ak\_survey\_id | links |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2,002 | AI | Aleutian Islands Bottom Trawl Survey | 5.2E+001 | 2.00201E+005 | 6.0E+000 | 7.22E+002 | 307-63 | Vesteraalen | 9.4E+001 | 05/17/2002 18:56:58 | 5.3737209999999997E+001 | -1.6701570000000001E+002 | 9.502E+004 | feathery bryozoan | Eucratea loricata | Low | 1.7493999999999999E-002 | 1.7494449999999999E+000 | 1.7494451079999999E+003 |  |  |  | 4.3999999999999997E-002 | 0 | 4.0999999999999996E+000 | 5.2999999999999998E+000 | 1.87E+002 | 1.5609999999999999E+000 | 1.6111999999999998E+001 | 7.25E+000 | 2.5150831999999994E+000 | 2.8000000000000003E-001 | 155,809 | 878,821 | [[data.frame]] |
| 2,002 | AI | Aleutian Islands Bottom Trawl Survey | 5.2E+001 | 2.00201E+005 | 6.0E+000 | 7.22E+002 | 307-63 | Vesteraalen | 9.4E+001 | 05/17/2002 18:56:58 | 5.3737209999999997E+001 | -1.6701570000000001E+002 | 7.9E+004 | squid unid. | Decapodiformes | High | 2.2266000000000001E-002 | 2.2265670000000002E+000 | 2.2265665009999998E+003 | 3.180809E+000 | 3.1808092900000003E+002 | 3.1808092869500001E+005 | 5.6000000000000001E-002 | 8.0E+000 | 4.0999999999999996E+000 | 5.2999999999999998E+000 | 1.87E+002 | 1.5609999999999999E+000 | 1.6111999999999998E+001 | 7.25E+000 | 2.5150831999999994E+000 | 2.8000000000000003E-001 |  | 878,822 | [[data.frame]] |
| 2,002 | AI | Aleutian Islands Bottom Trawl Survey | 5.2E+001 | 2.00201E+005 | 6.0E+000 | 7.22E+002 | 307-63 | Vesteraalen | 9.4E+001 | 05/17/2002 18:56:58 | 5.3737209999999997E+001 | -1.6701570000000001E+002 | 2.4191E+004 | shortfin eelpout | Lycodes brevipes | High | 3.5784000000000003E-002 | 3.5784099999999999E+000 | 3.5784104480000001E+003 | 7.9520199999999996E-001 | 7.9520231999999993E+001 | 7.9520232174000004E+004 | 8.9999999999999997E-002 | 2.0E+000 | 4.0999999999999996E+000 | 5.2999999999999998E+000 | 1.87E+002 | 1.5609999999999999E+000 | 1.6111999999999998E+001 | 7.25E+000 | 2.5150831999999994E+000 | 2.8000000000000003E-001 | 165,258 | 878,823 | [[data.frame]] |

## 12.2 Ex. 2: Load the first 10000 rows of data

# Not run because too big:  
res <- httr::GET(url = paste0(api\_link, "?offset=0&limit=10000"))  
data <- jsonlite::fromJSON(base::rawToChar(res$content))  
print(paste0("rows: ", dim(data$items)[1], "; cols: ", dim(data$items)[2]))

[1] "rows: 10000; cols: 36"

## 12.3 Ex. 3: Filter by Year

Show all the data greater than the year 2020.

res <- httr::GET(url = paste0(api\_link, '?q={"year":{"$gt":2020}}'))  
data <- jsonlite::fromJSON(base::rawToChar(res$content))  
flextable::flextable(  
 data$items[1:3, c("year", "srvy", "stratum", "species\_code", "cpue\_kgkm2")]) %>%   
 flextable::theme\_zebra()

| **year** | **srvy** | **stratum** | **species\_code** | **cpue\_kgkm2** |
| --- | --- | --- | --- | --- |
| 2,022 | AI | 7.22E+002 | 1.0261E+004 | 6.7332582200000002E+002 |
| 2,022 | AI | 7.93E+002 | 8.054E+004 | 3.6112E-001 |
| 2,022 | AI | 7.22E+002 | 2.1347E+004 | 7.5809130500000003E+002 |

## 12.4 Ex. 4: Filter by species name

Show all the data where the product name contains pollock Please note that here the word pollock is case sensitive.

The notation for finding a string is to use % around it. Since % is a reserved character in a URL, you have to replace % with %25.

res <- httr::GET(  
 url = paste0(api\_link, '?q={"common\_name":{"$like":"%25pollock%25"}}'))  
data <- jsonlite::fromJSON(base::rawToChar(res$content))  
flextable::flextable(  
 data$items[1:3, c("year", "srvy", "stratum", "species\_code", "cpue\_kgkm2")]) %>%   
 flextable::theme\_zebra()

| **year** | **srvy** | **stratum** | **species\_code** | **cpue\_kgkm2** |
| --- | --- | --- | --- | --- |
| 2,002 | AI | 7.21E+002 | 2.174E+004 | 6.3989099999999999E-001 |
| 2,002 | AI | 7.22E+002 | 2.174E+004 | 7.7532226400000002E+002 |
| 2,002 | AI | 7.22E+002 | 2.174E+004 | 1.0685806397E+004 |

## 12.5 Ex. 5: Combination of year and name filters

Show all the data where years > 2020 and the product name contains pollock

res <- httr::GET(  
 url = paste0(api\_link,   
 '?q={"year":{"$gt":2020},"common\_name":{"$like":"%25pollock%25"}}'))  
data <- jsonlite::fromJSON(base::rawToChar(res$content))  
flextable::flextable(  
 data$items[1:3, c("year", "srvy", "stratum", "species\_code", "cpue\_kgkm2")]) %>%   
 flextable::theme\_zebra()

| **year** | **srvy** | **stratum** | **species\_code** | **cpue\_kgkm2** |
| --- | --- | --- | --- | --- |
| 2,022 | AI | 7.22E+002 | 2.174E+004 | 2.2754334435000001E+004 |
| 2,022 | AI | 7.93E+002 | 2.174E+004 | 7.8536315350000004E+003 |
| 2,022 | AI | 7.21E+002 | 2.174E+004 | 7.2350103259999996E+003 |

## 12.6 Ex. 6: Combination of year, srvy, stratum

Show all the data where year = 1989, srvy = “EBS”, and stratum is not equal to 81

res <- httr::GET(  
 url = paste0(api\_link, '?q={"year":1989,"srvy":"EBS","stratum":{"$ne":"81"}}'))  
data <- jsonlite::fromJSON(base::rawToChar(res$content))  
flextable::flextable(  
 data$items[1:3, c("year", "srvy", "stratum", "species\_code", "cpue\_kgkm2")]) %>%   
 flextable::theme\_zebra()

| **year** | **srvy** | **stratum** | **species\_code** | **cpue\_kgkm2** |
| --- | --- | --- | --- | --- |
| 1,989 | EBS | 1.0E+001 | 4.05E+004 | 9.6200360000000007E+000 |
| 1,989 | EBS | 1.0E+001 | 6.8578E+004 | 9.6200360000000007E+000 |
| 1,989 | EBS | 1.0E+001 | 2.1313E+004 | 1.8179039E+001 |

## 12.7 Ex. 7: Visualize CPUE data in distribution map

Pacific cod catch-per-unit-effort estimates for NBS in 2021 and map constructed using [akgfmaps](https://github.com/afsc-gap-products/akgfmaps).

# res <- httr::GET(  
# url = paste0(api\_link, "?offset=0&limit=10000"),   
# query = list(year = 2021, srvy = "EBS", species\_code = 30060))  
res <- httr::GET(  
 url = paste0(api\_link, '?q={"year":2021,"srvy":"NBS","species\_code":21720}'))  
data\_catch <- jsonlite::fromJSON(base::rawToChar(res$content))$items %>%   
 dplyr::select(stratum, station, cpue\_kgkm2)   
  
# zero-fill data (imperfectly, but effective for this example)  
res <- httr::GET(  
 url = paste0(api\_link, '?q={"year":2021,"srvy":"NBS"}offset=0&limit=10000'))  
data\_haul <- jsonlite::fromJSON(base::rawToChar(res$content))$items %>%   
 dplyr::select(stratum, station, latitude\_dd, longitude\_dd) %>%  
 dplyr::distinct()  
  
data <- dplyr::left\_join(data\_haul, data\_catch) %>%   
 dplyr::mutate(cpue\_kgkm2 = ifelse(is.na(cpue\_kgkm2), 0, cpue\_kgkm2),   
 dplyr::across(dplyr::everything(), as.numeric))   
  
flextable::flextable(data[1:3,]) %>%   
 flextable::theme\_zebra()

| **stratum** | **station** | **latitude\_dd** | **longitude\_dd** | **cpue\_kgkm2** |
| --- | --- | --- | --- | --- |
| 71 |  | 63.70028 | -171.0225 | 1.183039 |
| 81 |  | 61.68600 | -173.0776 | 13,256.716473 |
| 81 |  | 61.34965 | -172.2251 | 600.958261 |

# devtools::install\_github("afsc-gap-products/akgfmaps", build\_vignettes = TRUE)  
library(akgfmaps)  
  
figure <- akgfmaps::make\_idw\_map(  
 CPUE\_KGHA = data$cpue\_kgkm2, # calculates the same, regardless of units.   
 LATITUDE = data$latitude\_dd,   
 LONGITUDE = data$longitude\_dd,   
 region = "bs.north", # Predefined EBS area  
 set.breaks = "jenks", # Gets Jenks breaks from classint::classIntervals()  
 in.crs = "+proj=longlat", # Set input coordinate reference system  
 out.crs = "EPSG:3338", # Set output coordinate reference system  
 grid.cell = c(20000, 20000), # 20x20km grid  
 key.title = "Pacific Ocean perch") # Include in the legend title

[inverse distance weighted interpolation]  
[inverse distance weighted interpolation]

figure$plot +   
 ggplot2::guides(fill=guide\_legend(title = "Pacific cod\nCPUE (kg/km2)"))

|  |
| --- |
| Ex. 7: Visualize CPUE data in distribution map. |

# 13. Access public data using the API and Python

### 13.0.1 {afscgap} Library Installation

author: Sam Pottinger (sam.pottinger@berkeley.edu; GitHub::sampottinger) date: May 13, 2023

The third-party afscgap Python package interfaces with FOSS to access AFSC GAP data. It can be installed via pip:

#The reticulate package provides a comprehensive set of tools for interoperability between Python and R.   
library(reticulate)

pip install afscgap  
pip install git+https://github.com/SchmidtDSE/afscgap.git@main

For more information on installation and deployment, see the [library documentation](https://pyafscgap.org).

### 13.0.2 Basic query

This first example queries for Pacific glass shrimp (*Pasiphaea pacifica*) in the Gulf of Alaska in 2021. The library will automatically generate HTTP queries, converting from Python types to [ORDS](https://www.oracle.com/database/technologies/appdev/rest.html) query syntax.

import afscgap  
  
query = afscgap.Query()  
query.filter\_year(eq=2021)  
query.filter\_srvy(eq='GOA')  
query.filter\_scientific\_name(eq='Pasiphaea pacifica')  
  
results = query.execute()

The results variable in this example is an iterator that will automatically perform pagination behind the scenes.

### 13.0.3 Iterating with a for loop

The easiest way to interact with results is a simple for loop. This next example determines the frequency of different catch per unit effort where Pacific glass shrimp were reported:

import afscgap  
  
# Mapping from CPUE to count  
count\_by\_cpue = {}  
  
# Build query  
query = afscgap.Query()  
query.filter\_year(eq=2021)  
query.filter\_srvy(eq='GOA')  
query.filter\_scientific\_name(eq='Pasiphaea pacifica')  
results = query.execute()  
  
# Iterate through results and count  
for record in results:  
 cpue = record.get\_cpue\_weight(units='kg/ha')  
 cpue\_rounded = round(cpue)  
 count = count\_by\_cpue.get(cpue\_rounded, 0) + 1  
 count\_by\_cpue[cpue\_rounded] = count  
  
# Print the result  
print(count\_by\_cpue)

Note that, in this example, only records with Pacific glass shrimp are included (“presence-only” data). See zero catch inference below. In other words, it reports on CPUE only for hauls in which Pacific glass shrimp were recorded, excluding some hauls like those in which Pacific glass shrimp were not found at all.

### 13.0.4 Iterating with functional programming

A for loop is not the only option for iterating through results. List comprehensions and other functional programming methods can be used as well.

import statistics  
  
import afscgap  
  
# Build query  
query = afscgap.Query()  
query.filter\_year(eq=2021)  
query.filter\_srvy(eq='GOA')  
query.filter\_scientific\_name(eq='Pasiphaea pacifica')  
results = query.execute()  
  
# Get temperatures in Celsius  
temperatures = [record.get\_bottom\_temperature(units='c') for record in results]  
  
# Take the median  
print(statistics.median(temperatures))

This example reports the median temperature in Celcius for when Pacific glass shrimp was reported.

### 13.0.5 Load into Pandas

The results from the afscgap package are serializable and can be loaded into other tools like [Pandas](https://pandas.pydata.org/). This example loads Pacific glass shrimp from 2021 Gulf of Alaska into a data frame.

import pandas  
  
import afscgap  
  
query = afscgap.Query()  
query.filter\_year(eq=2021)  
query.filter\_srvy(eq='GOA')  
query.filter\_scientific\_name(eq='Pasiphaea pacifica')  
results = query.execute()  
  
pandas.DataFrame(results.to\_dicts())

Specifically, to\_dicts provides an iterator over a dictionary form of the data that can be read into tools like Pandas.

### 13.0.6 Advanced filtering

Queries so far have focused on filters requiring equality but range queries can be built as well.

import afscgap  
  
# Build query  
query = afscgap.Query()  
query.filter\_year(min\_val=2015, max\_val=2019) # Note min/max\_val  
query.filter\_srvy(eq='GOA')  
query.filter\_scientific\_name(eq='Pasiphaea pacifica')  
results = query.execute()  
  
# Sum weight  
weights = map(lambda x: x.get\_weight(units='kg'), results)  
total\_weight = sum(weights)  
print(total\_weight)

This example queries for Pacific glass shrimp data between 2015 and 2019, summing the total weight caught. Note that most users will likely take advantage of built-in Python to [ORDS](https://www.oracle.com/database/technologies/appdev/rest.html) query generation which dictates how the library communicates with the API service. However, users can provide raw ORDS queries as well using [manual filtering](https://pyafscgap.org/devdocs/afscgap.html#manual-filtering).

### 13.0.7 Zero-catch inference

Until this point, these examples use presence-only data. However, the afscgap package can infer negative or “zero catch” records as well.

import afscgap  
  
# Mapping from CPUE to count  
count\_by\_cpue = {}  
  
# Build query  
query = afscgap.Query()  
query.filter\_year(eq=2021)  
query.filter\_srvy(eq='GOA')  
query.filter\_scientific\_name(eq='Pasiphaea pacifica')  
query.set\_presence\_only(False) # Added to earlier example  
results = query.execute()  
  
# Iterate through results and count  
for record in results:  
 cpue = record.get\_cpue\_weight(units='kg/ha')  
 cpue\_rounded = round(cpue)  
 count = count\_by\_cpue.get(cpue\_rounded, 0) + 1  
 count\_by\_cpue[cpue\_rounded] = count  
  
# Print the result  
print(count\_by\_cpue)

This example revisits the earlier snippet for CPUE counts but set\_presence\_only(False) directs the library to look at additional data on hauls, determining which hauls did not have Pacific glass shrimp. This lets the library return records for hauls in which Pacific glass shrimp were not found. This can be seen in differences in counts reported:

| Rounded CPUE | Count with set\_presence\_only(True) | Count with set\_presence\_only(False) |
| --- | --- | --- |
| 0 kg/ha | 44 | 521 |
| 1 kg/ha | 7 | 7 |
| 2 kg/ha | 1 | 1 |

Put simply, while the earlier example showed CPUE counts for hauls in which Pacific glass shrimp were seen, this revised example reports for all hauls in the Gulf of Alaska in 2021.

### 13.0.8 More information

Please see the [API documentation](https://pyafscgap.org/devdocs/afscgap.html) for the Python library for additional details.

# 14. Access public data (otherwise available from FOSS) using R in Oracle (AFSC only)

If the user has access to the AFSC Oracle database, the user can use SQL developer to view and pull the FOSS public data directly from the RACEBASE\_FOSS Oracle schema.

### 14.0.1 Connect to Oracle from R

Many users will want to access the data from Oracle using R. The user will need to install the RODBC R package and ask OFIS (IT) connect R to Oracle. Then, use the following code in R to establish a connection from R to Oracle:

Here, the user can write in their username and password directly into the RODBC connect function. Never save usernames or passwords in scripts that may be intentionally or unintentionally shared with others. If no username and password is entered in the function, pop-ups will appear on the screen asking for the username and password.

#' Define RODBC connection to ORACLE  
 #'  
 #' @param schema default = 'AFSC'.   
 #'  
 #' @return oracle channel connection  
 #' @export  
 #'  
 #' @examples  
 #' # Not run  
 #' # channel <- oracle\_connect()  
oracle\_connect <- function(  
 schema='AFSC',   
 username = NULL,   
 passowrd = NULL){(echo=FALSE)  
   
 library("RODBC")  
 library("getPass")  
 if (is.null(username)) {  
 username <- getPass(msg = "Enter your ORACLE Username: ")  
 }  
 if (is.null(password)) {  
 password <- getPass(msg = "Enter your ORACLE Password: ")  
 }  
 channel <- RODBC::odbcConnect(  
 paste(schema),  
 paste(username),  
 paste(password),   
 believeNRows=FALSE)  
 return(channel)  
}  
  
channel <- oracle\_connect()

### 14.0.2 Ex. 1: Join data

To join these tables in Oracle, you may use a variant of the following code:

### 14.0.3 Ex. 2: Subset data

Once connected, pull and save (if needed) the tables into the R environment.

To pull a small subset of the data (especially since files like RACEBASE\_FOSS.FOSS\_CPUE\_ZEROFILLED are so big), use a variation of the following code. Here, we are pulling EBS Pacific cod from 2010 - 2021:

# Pull data  
a <- RODBC::sqlQuery(  
channel = channel,   
query =   
"SELECT \* FROM GAP\_PRODUCTS.FOSS\_CATCH cc  
JOIN GAP\_PRODUCTS.FOSS\_HAUL hh  
ON cc.HAULJOIN = hh.HAULJOIN  
WHERE SRVY = 'EBS'   
AND COMMON\_NAME = 'Pacific cod'   
AND YEAR >= 2010   
AND YEAR < 2021")  
# Save table to local directory  
write.csv(x = a, file = "RACEBASE\_FOSS-FOSS\_CPUE\_ZEROFILLED-ebs\_pcod\_2010-2020.csv")

# 15. Acknowledgments

# 16. Community Acknowledgments

We would like to thank the many communities of Alaska and their members who have helped contribute to this body of work. The knowledge, experiences, and insights have been instrumental in expanding the scope of our science and knowledge to encompass the many issues that face this important ecosystem. We appreciate feedback from those residing in the region that are willing to share their insights and participation in an open dialog about how we can improve our collective knowledge of the ecosystem and the region.

# 17. Technical Acknowledgments

This quarto book is based off the [NOAA-quarto-book](https://github.com/nmfs-opensci/NOAA-quarto-book) GitHub repo designed by Eli Holmes.

This repo and GitHub Action was based on the tutorial by Openscapes [quarto-website-tutorial](https://github.com/Openscapes/quarto-website-tutorial) by Julia Lowndes and Stefanie Butland.

## 17.1 Partners

Scientists from the Alaska Fisheries Science Center conduct these bottom trawl surveys with participation from the Alaska Department of Fish & Game (ADF&G), the International Pacific Halibut Commission (IPHC), and universities. This research is conducted on chartered fishing vessels.

# 18. Production Run Notes

# 19. R Version Metadata

R version 4.3.0 (2023-04-21 ucrt)  
Platform: x86\_64-w64-mingw32/x64 (64-bit)  
Running under: Windows 10 x64 (build 19045)  
  
Matrix products: default  
  
  
locale:  
[1] LC\_COLLATE=English\_United States.utf8   
[2] LC\_CTYPE=English\_United States.utf8   
[3] LC\_MONETARY=English\_United States.utf8  
[4] LC\_NUMERIC=C   
[5] LC\_TIME=English\_United States.utf8   
  
time zone: America/Los\_Angeles  
tzcode source: internal  
  
attached base packages:  
[1] stats graphics grDevices utils datasets methods base   
  
loaded via a namespace (and not attached):  
 [1] compiler\_4.3.0 fastmap\_1.1.1 cli\_3.6.1 tools\_4.3.0   
 [5] htmltools\_0.5.5 rstudioapi\_0.15.0 yaml\_2.3.7 rmarkdown\_2.23   
 [9] knitr\_1.43 jsonlite\_1.8.7 xfun\_0.39 digest\_0.6.33   
[13] rlang\_1.1.1 evaluate\_0.21

### 19.0.1 NOAA README

This repository is a scientific product and is not official communication of the National Oceanic and Atmospheric Administration, or the United States Department of Commerce. All NOAA GitHub project code is provided on an ‘as is’ basis and the user assumes responsibility for its use. Any claims against the Department of Commerce or Department of Commerce bureaus stemming from the use of this GitHub project will be governed by all applicable Federal law. Any reference to specific commercial products, processes, or services by service mark, trademark, manufacturer, or otherwise, does not constitute or imply their endorsement, recommendation or favoring by the Department of Commerce. The Department of Commerce seal and logo, or the seal and logo of a DOC bureau, shall not be used in any manner to imply endorsement of any commercial product or activity by DOC or the United States Government.

### 19.0.2 NOAA License

Software code created by U.S. Government employees is not subject to copyright in the United States (17 U.S.C. §105). The United States/Department of Commerce reserve all rights to seek and obtain copyright protection in countries other than the United States for Software authored in its entirety by the Department of Commerce. To this end, the Department of Commerce hereby grants to Recipient a royalty-free, nonexclusive license to use, copy, and create derivative works of the Software outside of the United States.

# References

Hoff, G. R. (2016). *Results of the 2016 eastern Bering Sea upper continental slope survey of groundfishes and invertebrate resources* (NOAA Tech. Memo. NOAA-AFSC-339). U.S. Dep. Commer. <https://doi.org/10.7289/V5/TM-AFSC-339>

Markowitz, E. H., Dawson, E. J., Anderson, A. B., Rohan, S. K., Charriere, N. E., Prohaska, B. K., and Stevenson, D. E. (2023). *Results of the 2022 eastern and northern Bering Sea continental shelf bottom trawl survey of groundfish and invertebrate fauna* (NOAA Tech. Memo. NMFS-AFSC-469; p. 213). U.S. Dep. Commer.

Von Szalay, P. G., and Raring, N. W. (2018). *Data report: 2017 Gulf of Alaska bottom trawl survey* (NOAA Tech. Memo. NMFS-AFSC-374). U.S. Dep. Commer. <https://doi.org/10.7289/V5/TM-AFSC-374>

Von Szalay, P. G., and Raring, N. W. (2020). *Data report: 2018 Aleutian Islands bottom trawl survey* (NOAA Tech. Memo. NMFS-AFSC-409). U.S. Dep. Commer. <https://doi.org/10.25923/qe5v-fz70>

# 20. Contact us

**General questions and more specific data requests** can be sent to [afsc.gap.metadata@noaa.gov](mailto:afsc.gap.metadata@noaa.gov) or submitted as an [issue on our GitHub Organization](https://github.com/afsc-gap-products/data-requests). The version of this data used for stock assessments can be found through the Alaska Fisheries Information Network (AKFIN). For questions about the eastern Bering Sea surveys, contact Duane Stevenson ([Duane.Stevenson@noaa.gov](mailto:Duane.Stevenson@noaa.gov)). For questions about the Gulf of Alaska or Aleutian Islands surveys, contact Ned Laman ([Ned.Laman@noaa.gov](mailto:Ned.Laman@noaa.gov)). For questions specifically about crab data in any region, contact Mike Litzow ([Mike.Litzow@noaa.gov](mailto:Mike.Litzow@noaa.gov)), the Shellfish Assessment Program lead.

For questions, comments, and concerns specifically about the [Fisheries One Stop Shop (FOSS)](https://www.fisheries.noaa.gov/foss) platform, please contact us using the Comments page on the [FOSS](https://www.fisheries.noaa.gov/foss) webpage.

Alaska Fisheries Science Center (AFSC)  
National Oceanic and Atmospheric Administration (NOAA)  
Resource Assessment and Conservation Engineering Division (RACE)  
Groundfish Assessment Program (GAP)  
7600 Sand Point Way, N.E. bldg. 4  
Seattle, WA 98115 USA

## 20.1 Suggestions and comments

If the data or metadata can be improved, please create a pull request, [submit an issue to the GitHub organization](https://github.com/afsc-gap-products/data-requests/issues) or [submit an issue to the code’s repository](https://github.com/afsc-gap-products/gap_products/issues).