



# NOAA FISHERIES

## **Electronic Vessel Trip Reporting (eVTR) Technical Requirements Guide**

For eVTR Application Developers

Rev. 2.0

Greater Atlantic Regional Fisheries Office (GARFO)

*Analysis & Program Support Division (APSD)*

*Gloucester, MA*

*3/27/2024*



## Table of Contents

1	Introduction .....	5
1.1	Audience .....	5
1.2	Product Support.....	5
1.3	Note Concerning Revision History .....	5
2	eVTR Composition.....	6
2.1	How GARFO Organizes eVTR Information .....	6
2.2	Subtrips .....	7
2.3	eVTR Unique Trip Identifier (UTID).....	7
2.4	File Format .....	8
2.5	Electronic Signature .....	9
2.6	File Transfer .....	9
2.7	Error Checking.....	9
2.8	eVTR User Interface Requirements .....	10
3	JSON API Requirements .....	11
3.1	JSON API Generic Sequence.....	11
3.2	API Authentication Protocol .....	12
3.3	Send eVTR Form to GARFO .....	14
4	eVTR Reference.....	17
4.1	Support Files .....	17
4.2	eVTR Form Specification .....	18
4.3	Mandatory Use of a Field in eVTR Based on Trip Activity Type.....	20
4.4	Gear Code and Soak Time Interdependence .....	22
4.5	Quality Control Audit Rules .....	26
4.6	Quality Control Audit Rules for Lobster and Crab .....	30
4.7	Regulatory References.....	31
5	Requirements for Lobster Reporting (Report_Type=2).....	32
5.1	Report Type Field .....	32
5.2	Trip Type Label.....	32
5.3	New Trip-level Input Fields for Lobster (Report_Type=2) .....	33
5.4	Effort-level Updates for Lobster (Report_Type=2) .....	34
5.5	Existing Effort-level Fields for Lobster (Report_Type=2).....	36
5.6	Removed Effort-level Fields for Lobster (Report_Type=2).....	36
5.7	Gear Options for Lobster (Report_Type=2) .....	37
5.8	Endpoint.....	37
5.9	Example JSONs.....	37
6	Requirements for Clam Reporting (Report_Type=3).....	45
6.1	Report Type Field .....	45
6.2	Catch-level Changes for Clams (Report_Type=3) .....	47
6.3	New Disposition Input Fields for Clams (Report Type = 3) .....	48
6.4	JSON Example for Report_Type=3 .....	49
7	Document Revision History.....	51
7.1	Revision 1.0 Updates .....	51
7.2	Revision 1.1 Updates .....	51
7.3	Revision 1.2 Updates .....	51
7.4	Revision 1.2.1 Updates .....	52

7.5 Revision 1.3 Updates ..... 52

7.6 Revision 1.4 Updates ..... 53

7.7 Revision 1.4.1 Updates ..... 54

7.8 Revision 1.5 Updates ..... 54

7.9 Revision 2.0 Updates ..... 54

# 1 Introduction

This guide provides the technical information that you need to satisfy the programming requirements for electronic reporting. The guide comprises three principal sections:

- The basics of GARFO's Electronic Vessel Trip Report (eVTR)
- Application Programming Interface (API) requirements for software application developers

---

**NOTE:** The Federal Register outlines the regulatory requirements that eVTR software must meet. See the proper references in the section [Regulatory References](#) in this guide.

---

- Reference information

Use the links in the Table of Contents or the links embedded throughout this guide to quickly navigate to the information you want.

## 1.1 Audience

This guide is intended for programmers whose expertise ranges from intermediate to advanced. This guide is not intended for novice programmers who do not have a working knowledge of APIs. Novice programmers should refer to API tutorials. More experienced users should feel free to use the Table of Contents in order to look up specific information as needed.

## 1.2 Product Support

GARFO must endorse your product's adherence to regulatory and technical requirements. To request assistance, contact the IT Service Desk by phone or by e-mail:

- Phone: 978-281-9188
- E-mail: [nmfs.gar.helpdesk@noaa.gov](mailto:nmfs.gar.helpdesk@noaa.gov)

## 1.3 Note Concerning Revision History

When GARFO changes eVTR technical requirements, we also update the information in this guide to reflect those changes. To obtain important information about updates to the technical requirements at this release, or to obtain information about updates at previous releases, please consult the section [Document Revision History](#) elsewhere in this guide.

## 2 eVTR Composition

The fishing vessel's owner/operator files a Vessel Trip Report (VTR) after a fishing trip. If the owner/operator files the report electronically, the report is called an eVTR. The report includes but is not limited to information about the trip duration, the fishing effort expended (gear used, location, and duration fished), the species that were caught, and their landed weight. VTR data are the definitive source of fishing effort information (gear and area) which is a critical input to ensure accurate catch monitoring.

Information in this section includes the following topics:

- [How GARFO Organizes eVTR Information](#)
- [Subtrips](#)
- [eVTR Unique Trip Identifier \(UTID\)](#)
- [File Format](#)
- [Electronic Signature](#)
- [File Transfer](#)
- [Error Checking](#)
- [eVTR User Interface Requirements](#)

### 2.1 How GARFO Organizes eVTR Information

GARFO internally organizes the fishing trip data in an eVTR as follows:

#### 1) Trip

Identifies the vessel, trip type (commercial, party, charter or recreational), number of crew/anglers and the dates and times for when it departed and returned to port (sailed and landed respectively).

#### 2) Effort

Identifies subtrip information (see the section [Subtrips](#)). This portion of the VTR captures the location where the vessel fished, gear information, haul/soak time, number of hauls/sets/strings, etc.

#### 3) Catch

Identifies the species and the amount of the catch (*kept* and *discarded*) for each subtrip.

#### 4) Sales

Identifies the dealer(s) that bought the catch (or the ultimate disposition of the catch), and also the date and port of the offload associated with each subtrip.

#### 5) Signature

Provides the verification by the vessel owner / operator that the information being submitted on the report is correct.

[Jump to: Table of Contents](#)

## 2.2 Subtrips

A subtrip represents the effort and catch associated with a unique combination of the following:

- Chart area
- Gear type
- Mesh size

If any of these elements change during the course of a trip, a new subtrip starts, and the vessel owner/operator must record additional catch and effort information.

A well-designed eVTR program or application has the ability to significantly reduce the amount of time required for a vessel operator to comply with their VTR reporting requirements by eliminating the need to fill out information that stays the same throughout the trip (for example, vessel permit, date sail and land, trip type).

This means that the application programmer must code the application so that the user supplies data for the fields in the Trip and Signature information only once, since these fields apply to the entire trip.

## 2.3 eVTR Unique Trip Identifier (UTID)

The eVTR application must build and generate a unique trip identifier (UTID) that serves as the key by which all fisheries data systems are referenced from a specific fishing trip.

---

**CAUTION:** The UTID must be unique for every fishing trip; different trips can never use the same UTID. Duplicate UTIDs cause significant problems with data verification and will affect end users, software vendors, and GARFO.

---

### 2.3.1 UTID Composition

The eVTR application must build the UTID from the following numeric data:

- Vessel Permit Number (6 digits) as listed on the vessel's Federal fishing permit

The Vessel Permit Number is also referred to as the VPN.

- Year (2 digits)
- Month (2 digits)
- Day (2 digits)
- Hour (2 digits, based upon 24 hour clock / military time)

[Jump to: Table of Contents](#)

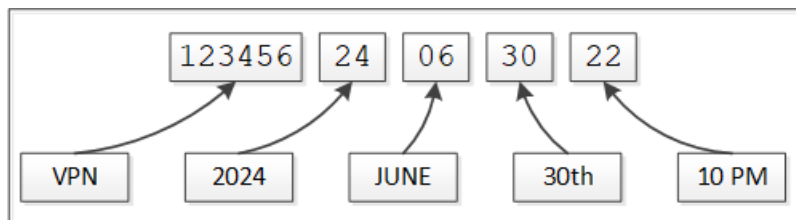
## 2.3.2 UTID Example

A single numeric string constitutes the eVTR UTID, such as the following example:

12345621063023

[Figure 1](#) shows how to read the eVTR UTID:

Figure 1: Reading the eVTR UTID



---

**NOTE:** The year, month, day and hour portion of the eVTR UTID can represent the date/time at which the trip report was initiated, the date/time the trip started, or anywhere in between these two dates/times.

---

## 2.3.3 UTID Usage Notes

Be aware of the following information in accounting for the eVTR UTID:

- Once the eVTR application generates the value of the eVTR UTID, the value cannot change.
- The eVTR UTID shall be readily available to the vessel owner/operator. The UTID:
  - Must be available when the trip report is initiated.
  - Is required by the seafood dealer before the catch can be purchased from the vessel.
- If two duplicate trip IDs are created by starting trips in rapid succession, it is acceptable to increment the last two digits.
- NMFS uses the eVTR UTID to link a fishing trip to other Fisheries Dependent Data (FDD) systems that collect and process data, such as Dealer reports and Observer reports.

## 2.4 File Format

An eVTR application must fulfill the following file format requirements:

- 1) Produce a data file in the specified standard format.

Such a format allows a successful file upload into the GARFO databases. Find the valid file format in the section [eVTR Form Specification](#) in this guide.

- 2) Meet GARFO data collection requirements.

GARFO provides support tables for standard industry information that a programmer needs to build into the application. Find a list of the valid support tables in the section labeled [Support](#) in this guide.

---

**NOTE:** A value of **NA** in a table cell indicates that there is no value or instruction from GARFO concerning that particular instance.

---

[Jump to: Table of Contents](#)



## 2.5 *Electronic Signature*

Immediately upon completion of a trip report, the vessel owner/operator must have the means to digitally sign the eVTR. The eVTR application must provide the following for electronic signature:

- A way for the vessel owner/operator to verify the operator name and Vessel Permit Number (VPN).
- The display of the following text during the signing ceremony:

```
"I certify that the vessel trip information I am providing is true, complete
and correct to the best of my knowledge, and made in good faith. Making a false
statement is punishable by law (18 U.S.C. 1001)."
```

## 2.6 *File Transfer*

GARFO requires that you submit trip report files using the HTTPS POST method. Use the API URL to submit the eVTR. For information, refer to the section [JSON API Requirements](#) in this guide.

## 2.7 *Error Checking*

When the vessel owner/operator submits a trip report to GARFO, the audit engine checks the report against a set of audit rules. The audit engine flags any eVTR that fails one or more audit rules. GARFO subsequently instructs the vessel owner/operator to first correct, then resubmit the report.

Reports submitted through your application will pass through the same audit engine checks. The checks may also be built into your application in order to prevent the rejection of the report once it is submitted to GARFO. Find a list of the GARFO QC audits in the section [Quality Control Audit Rules](#) later in this guide.

### 2.7.1 *Corrections to a Previously Submitted eVTR*

GARFO can identify most syntax errors in a submitted eVTR before processing that eVTR. However, GARFO can only identify other types of errors, such as discrepancies between VTR and dealer data sets, after processing the eVTR in the audit rules engine.

If the audit returns one or more errors against the submission, GARFO notifies the vessel owner/operator of the problem(s) so that they may correct the eVTR, and then resubmit. This action requires that eVTR vendors provide a method for copying and then modifying, sometimes significantly, a previously submitted eVTR. The vessel owner/operator must then have the ability to submit the amended eVTR in the same manner as the original submission.

---

**CAUTION:** The vessel owner/operator **must** use the trip identifier from the original submission when submitting the revised eVTR.

---

[Jump to: Table of Contents](#)

## 2.8 *eVTR User Interface Requirements*

The eVTR application shall adhere to the following requirements for user interface design:

- Record fishing data accurately as entered.
- Transmit data to GARFO that is an accurate representation of what was reported by the vessel owner/operator.
- Store data entered by owners/operators even if no internet connection is available.
- Enforce the submission of a valid value for each data element required on a particular fishing trip.
- Allow the owner/operator to enter, choose, or confirm all data elements.
- Prevent data element auto-population, or default value generation, without confirmation.
- Generate a unique trip identifier (UTID) as soon as a new trip report is initiated.
- Provide the ability to display a facsimile of all previously submitted reports submitted on that device.

When requested by authorized personnel, eVTR applications must have the capability to display a facsimile of the paper VTR form, including a separate 'page' for each subtrip. Ideally, the eVTR software should enable print, export, or email VTR facsimiles.

- Allow the manual entry of latitude/longitude coordinates while also optionally supporting the coordinates to be automatically populated from the device or vessel's GPS system.

[Jump to: Table of Contents](#)

## 3 JSON API Requirements

GARFO uses a JavaScript Object Notation (JSON) based API for 3<sup>rd</sup>-party communications with the eVTR server. JSON is a lightweight, open standard data exchange format, uses a natural language coding scheme, and can work with any framework or platform. Topics in this section include:

- [JSON API Generic Sequence](#)
- [API Authentication Protocol](#)
- [Send eVTR Form to GARFO](#)

### 3.1 JSON API Generic Sequence

Data is passed and returned in JSON format over the HTTPS protocol by means of requests and responses. You code a request to obtain a resource that's stored on a remote server, or to transmit data to a remote server, and the remote server responds with the requested information or message. Each action requires a separate request and response.

The following sequence describes a generic request and return exchange between the 3<sup>rd</sup>-party application and the database server:

- 1) A 3<sup>rd</sup>-party application submits an authentication request (if needed) from the device to the API by means of a URL.
- 2) The application receives a server STATUS message as a JSON object.

If authentication is successful, that response will include a token string.

- 3) The bundle of the request (for example, an eVTR form) is passed in the POST header of the HTTPS request using the JSON object format.
- 4) The API returns the requested data or response as a JSON object.

The return includes a STATUS indicator.

#### 3.1.1 Base URL

The base URL link is as follows:

[https://apps-garfo.fisheries.noaa.gov/base\\_api/](https://apps-garfo.fisheries.noaa.gov/base_api/)

This URL forms the basis of all requests to the eVTR API module. Create specific requests by appending request strings on this URL.

#### 3.1.2 JSON API Usage Notes

Be aware of the following before you begin to use the GARFO eVTR API module:

- All errors return a STATUS as a JSON object that contains a description of the error.  
See the section named [Quality Control Audit Rules](#) for a list of valid codes.

[Jump to: Table of Contents](#)

## 3.2 API Authentication Protocol

This section describes how to properly submit an authentication request to the system.

### 3.2.1 Prerequisite: Obtain Valid Username and Password

A Federally-permitted vessel owner / operator must obtain a valid username / password combination from the IT Service Desk **before** being able to submit a VTR electronically:

- IT Service Desk phone: 978-281-9188
- IT Service Desk e-mail: [nmfs.gar.helpdesk@noaa.gov](mailto:nmfs.gar.helpdesk@noaa.gov)

### 3.2.2 Authentication

The API authenticates the user's credentials and returns an alphanumeric token that represents the validation of credentials. The token is valid for 45 minutes after it is issued, and may be included in a VTR within that time period.

- The authentication values are in the submitted JSON.
- The order of the fields does not matter.

```
{
  "authorization":
  {
    "username": "Thetis",
    "password": "Thetis_password",
    "vendor_id": "1234567",
    "vendor_software_version": "1.1.0.1",
    "vendor_app_name": "ReallyNeat FishingApp",
    "vendor_app_platform": "ANDROID"
  }
}
```

#### 3.2.2.1 App Platform

The field `vendor_app_platform` key should describe the operating system or environment that the app is running on. This might be used to identify and remedy an issue in a certain version of an app, or if the codebase is shared between platforms, it is not necessary to identify the platform to fill out this field. For example, a JavaScript app that runs on multiple platforms could use `BROWSER`. Recommended values:

- BROWSER
- IOS
- IPAD
- IPHONE
- MAC CATALYST
- MAC OS
- ANDROID
- WINDOWS

Or any value deemed reasonable for your application.

[Jump to: Table of Contents](#)

### 3.2.3 Returns from Authentication Request

Authentication attempt returns one of two possible results.

- Return from correct authorization
- Return from incorrect authorization

#### 3.2.3.1 *Return from Correct Authorization*

Following is a representation of the return from a correct authorization request. The values themselves are for example purposes only.

```
{
  "user_app_status": "TEST",
  "user_full_name": "user_full_name",
  "Status": "Logged In",
  "token": "a_very_long_cryptic_string",
  "token_expires": 1609941713,
  "client_id": "id of user",
  "operator_key": "operator key"
  "vtr_preferences": [],
  "broadcast_message": [],
  "entitlements": [
    {
      "vessel": "555555"
    },
    {
      "vessel": "555556"
    }
  ]
}
```

#### 3.2.3.2 *Return from Incorrect Authorization*

Following is an example of the return from an incorrect authorization request.

```
{
  "Status": "Logged Out"
}
```

[Jump to: Table of Contents](#)

## 3.3 *Send eVTR Form to GARFO*

This call submits an eVTR form for a logged in, authenticated user to the GARFO eVTR database.

### 3.3.1 Request Format

Following is the JSON format. The values in the examples in this section are placeholders only. The order of the fields is your choice.

### 3.3.2 Endpoints

Following are the supported endpoints for each submission request. Be aware that adding the string `test` to the end of the URL directs the request to a test system, and does not affect the vessel owner/operator's reporting compliance in any way. Use the test endpoint to verify your code.

---

**WARNING: *DO NOT* submit test data to the production endpoint!**

---

[https://apps-garfo.fisheries.noaa.gov/base\\_api/evtr/v\\_1\\_sales](https://apps-garfo.fisheries.noaa.gov/base_api/evtr/v_1_sales) (Production)

[https://apps-garfo.fisheries.noaa.gov/base\\_api/evtr/v\\_1\\_sales/test](https://apps-garfo.fisheries.noaa.gov/base_api/evtr/v_1_sales/test) (Test)

[Jump to: Table of Contents](#)

### 3.3.3 JSON Format (Test)

---

**WARNING: *DO NOT* submit test data to the production endpoint!**

---

#### 3.3.3.1 URL

[https://apps-garfo.fisheries.noaa.gov/base\\_api/evtr/v\\_1\\_sales/test](https://apps-garfo.fisheries.noaa.gov/base_api/evtr/v_1_sales/test)

#### 3.3.3.2 Description

The user designs the eVTR submit request so that Sales and Catch are self-contained units of data. The separate arrays method means that you don't have to link Sales to a specific Effort.

---

**CAUTION:** The total of the *kept* value ***must equal*** the total *sold* value for the entire trip.

---

```
{
  "vendor_id": 22222,
  "vendor_software_version": "0.0.0",
  "vendor_app_name": "ReallyNeat FishingApp",
  "vendor_app_platform": "ANDROID",
  "token": "ABC123ABC123ABC123ABC123",
  "client_id": 99999,
  "vendor_message_id": "",
  "trip_id": 11111114091913,
  "vessel_permit_num": 111111,
  "vessel_registration": "11",
  "date_sail": 2109190739,
  "date_land": 2109190739,
  "trip_type": 1,
  "number_of_crew": 10,
  "number_of_anglers": null,
  "operator_first_name": "test",
  "operator_last_name": "operator",
  "operator_permit_number": 90000001,
  "no_effort_trip_reason": 0,
  "trip_comments": "",
  "subtrips": [{
    "gearcode": "403LLB",
    "mesh_size": "6",
    "gearqty": "120",
    "gearsizes": "1.2",
    "latitude_or_loran1": "42:17",
    "longitude_or_loran2": "-068:51",
    "chart_area": 521,
    "fishing_depth": 50,
    "fishing_depth_uom": "fa",
    "number_of_hauls": 4,
    "tow_soak_time": "5:30",
    "catch": [{
      "species": "cod",
```

```
        "kept": 1000,  
        "discarded": 100,  
        "uom": "pounds"  
    }]  
  }],  
  "sales": [{  
    "species": "cod",  
    "sold": 1000,  
    "uom": "pounds",  
    "dealer_permit_number": 1621,  
    "dealer_name": "a & a seafood inc",  
    "date_sold": 210920  
    "offloading_port": 220101  
  }]
```

[Jump to: Table of Contents](#)



## 4 eVTR Reference

Use this section to obtain reference information on eVTR format and content. Topics include:

- [Support](#)
- [eVTR Form Specification](#)
- [Quality Control Audit Rules](#)
- [Mandatory Use of a Field in eVTR Based on Trip Activity Type](#)
- [Gear Code and Soak Time Interdependence](#)
- [Regulatory References](#)

---

**NOTE:** A value of **NA** in a table cell indicates that there is no value or instruction from GARFO concerning that particular instance.

---

### 4.1 Support Files

The eVTR Support Files provide information and codes that you need to build your application.

The support JSON files can be found here:

- <https://apps-garfo.fisheries.noaa.gov/fo/assets/Dealers.json>  
Provides dealer permit number, dealer name, and dealer codes for Home Consumption, Bait, etc.
- <https://apps-garfo.fisheries.noaa.gov/fo/assets/Gear.json>  
Provides valid gear codes along with additional information that can be used for range checking values such as mesh size, quantity, soak time etc. associated with a given gear code.
- <https://apps-garfo.fisheries.noaa.gov/fo/assets/Port.json>  
Provides the port code for a specified port and state combination.
- <https://apps-garfo.fisheries.noaa.gov/fo/assets/Species.json>  
Provides synonym(s), species code and common name for species in GARFO. The synonyms provide for lookups using colloquial names. For example, DAB = American Plaice Flounder.
- <https://apps-garfo.fisheries.noaa.gov/fo/assets/Activity.json>  
Subcategories used to refine a trip activity type.
- <https://apps-garfo.fisheries.noaa.gov/fo/assets/TripTypes.json>  
Valid trip types
- <https://apps-garfo.fisheries.noaa.gov/fo/assets/ChartArea.json>  
Valid chart areas
- <https://apps-garfo.fisheries.noaa.gov/fo/assets/Operators.json>  
Operator Permits
- <https://apps-garfo.fisheries.noaa.gov/fo/assets/Permits.json>  
Vessel Permits

[Jump to: Table of Contents](#)

## 4.2 eVTR Form Specification

The information presented in [Table 1](#) provides the data specifications (such as data type and numeric precision) for each field used in the eVTR form.

**Table 1: Data Formats for eVTR Form**

FIELD NAME	TYPE	LENGTH OR PRECISION	PARENT	NULLABLE	FORMAT
<i>vendor_id</i>	INTEGER	NA	NA	NO	NA
<i>vendor_software_version</i>	VARCHAR	NA	NA	NO	NA
<i>vendor_app_platform</i>	VARCHAR	NA			
<i>token</i>	VARCHAR	500	NA	YES	NA
<i>client_id</i>	NUMBER	10	NA	NO	NA
<i>vendor_message_id</i>	INTEGER	150	NA	YES	NA
<i>trip_id</i>	INTEGER	14	NA	NO	Exactly 14 digits
<i>vessel_registration</i>	VARCHAR	10	NA	NO	NA
<i>date_sail</i>	DATE	12	NA	NO	YYMMDDHH24MI
<i>date_land</i>	DATE	12	NA	NO	YYMMDDHH24MI
<i>trip_type</i>	INTEGER	1	NA	NO	NA
<i>number_of_crew</i>	INTEGER	2	NA	NO	NA
<i>vessel_permit_num</i>	INTEGER	6	NA	NO	Exactly 6 digits
<i>number_of_anglers</i>	INTEGER	3	NA	YES	NA
<i>operator_first_name</i>	VARCHAR	17	NA	YES	NA
<i>operator_last_name</i>	VARCHAR	17	NA	YES	NA
<i>operator_permit_number</i>	INTEGER	8	NA	YES	NA
<i>no_effort_trip_reason</i>	INTEGER	1	NA	NO	NA
<i>trip_comments</i>	VARCHAR	150	NA	YES	NA
<i>subtrips</i>	Array	NA	NA	YES	NA
<i>gearcode</i>	VARCHAR	3	subtrips	NO	Exactly 3 characters
<i>mesh_size</i>	NUMBER	6.1	subtrips	YES	NA
<i>gearqty</i>	NUMBER	5	subtrips	YES	NA
<i>gearsizes</i>	NUMBER	6.1	subtrips	YES	NA

FIELD NAME	TYPE	LENGTH OR PRECISION	PARENT	NULLABLE	FORMAT
<i>latitude_or_loran1</i>	VARCHAR	8	subtrips	NO	(-)DD.DDDD or (-)DD:MM
<i>longitude_or_loran2</i>	VARCHAR	9	subtrips	NO	(-)DDD.DDDD or (-)DDD:MM
<i>chart_area</i>	NUMERIC STRING	3	subtrips	NO	NA
<i>fishing_depth</i>	INTEGER	4	subtrips	NO	NA
<i>fishing_depth_uom</i>	VARCHAR	3	subtrips	NO	NA
<i>number_of_hauls</i>	INTEGER	4	subtrips	YES	NA
<i>tow_soak_time</i>	VARCHAR	7	subtrips	YES	9999:59
<i>catch</i>	Array	NA	NA	YES	NA
<i>species</i>	VARCHAR	10	catch	NO	NA
<i>kept</i>	INTEGER	8	catch	YES	NA
<i>discarded</i>	INTEGER	8	catch	YES	NA
<i>uom</i>	VARCHAR	3	catch	YES	NA
<i>dealer_permit_number</i>	INTEGER	5	catch	YES	NA
<i>dealer_name</i>	VARCHAR	40	catch	YES	NA
<i>date_sold</i>	DATE	6	catch	YES	YYMMDD
<i>offloading_port</i>	NUMERIC STRING	6	catch	YES	NA

[Jump to: Table of Contents](#)

### 4.3 Mandatory Use of a Field in eVTR Based on Trip Activity Type

[Table 2](#) lists each field in the eVTR. An **X** indicates that the field must have a provided value, based on the trip activity type, as presented in the eVTR Trip Activity Types support table.

**NOTE:** A value of NA in a table cell indicates that there is no value or instruction from GARFO concerning that particular instance.

**Table 2: Reporting Fields Based on Trip Activity Type**

Field Name	MANDATORY FIELDS FOR TRIP ACTIVITY TYPE 0 (FISHING TRIP WITH EFFORT(S))	MANDATORY FIELDS FOR TRIP ACTIVITY TYPES 1-8
<i>trip_id</i>	<b>X</b>	<b>X</b>
<i>vessel_permit_num</i>	<b>X</b>	<b>X</b>
<i>vessel_registration</i>	<b>X</b>	<b>X</b>
<i>date_sail</i>	<b>X</b>	<b>X</b>
<i>date_land</i>	<b>X</b>	<b>X</b>
<i>trip_type</i>	<b>X</b>	<b>X</b>
<i>number_of_crew</i>	Mandatory when <i>trip_type</i> reported as 1 (commercial), 2 (party) or 3 (charter)	Mandatory when <i>trip_type</i> reported as 1 (commercial), 2 (party) or 3 (charter)
<i>number_of_anglers</i>	Mandatory when <i>trip_type</i> reported as 2 (party), 3 (charter) or 6 (private/recreational)	Mandatory when <i>trip_type</i> reported as 2 (party), 3 (charter) or 6 (private/recreational)
<i>operator_name</i>	<b>X</b>	<b>X</b>
<i>operator_permit_number</i>	NA	NA
<i>no_effort_trip_reason</i> (Trip Activity Type)	<b>X</b>	<b>X</b>
<i>gearcode</i>	<b>X</b>	NA
<i>mesh_size</i>	Mandatory depending on <i>gear_code</i> reported. See <a href="#">Table 3</a> in this guide.	NA
<i>gearqty</i>	Mandatory depending on <i>gear_code</i> reported. See <a href="#">Table 3</a> in this guide.	NA
<i>gearsizes</i>	Mandatory depending on <i>gear_code</i> reported. See <a href="#">Table 3</a> in this guide.	NA
<i>latitude_or_loran1</i>	<b>X</b>	NA
<i>longitude_or_loran2</i>	<b>X</b>	NA
<i>chart_area</i>	<b>X</b>	NA
<i>fishing_depth</i> (UOM = Fathoms)	<b>X</b>	NA

Field Name	MANDATORY FIELDS FOR TRIP ACTIVITY TYPE 0 (FISHING TRIP WITH EFFORT(S))	MANDATORY FIELDS FOR TRIP ACTIVITY TYPES 1-8
<i>number_of_hauls</i>	Mandatory depending on <i>gear_code</i> reported. See <a href="#">Table 3</a> in this guide.	NA
<i>tow_soak_time</i>	Mandatory but <i>tow_soak_time</i> reported may or may not exceed trip duration, depending upon <i>gear_code</i> reported. See <a href="#">Table 3</a> for requirements.	NA
<i>species</i>	<b>X</b>	NA
<i>kept</i>	Mandatory unless <i>species</i> reported is NC (no catch). If <i>species</i> value equals NC, <i>kept</i> can be NULL or 0.	NA
<i>discarded</i>	Mandatory when <i>kept</i> is NULL or 0 and <i>species</i> code is not NC (no catch). If <i>kept</i> is 1 or greater, the field can still be populated by the user.	NA
<i>uom</i>	Mandatory; <i>uom</i> for <i>trip_type</i> 1 = pounds, <i>uom</i> for <i>trip_type</i> 2, 3 and 6 = count.	NA
<i>dealer_permit_number</i>	Mandatory unless the value reported for <i>kept</i> is NULL or 0.	NA
<i>dealer_name</i>	Mandatory unless the value reported for <i>kept</i> is NULL or 0.	NA
<i>date_sold</i>	Mandatory unless the value reported for <i>dealer_permit_number</i> is 99998, 1, 4, or 8.	NA
<i>offloading_port</i>	<b>X</b>	NA

[Jump to: Table of Contents](#)

## 4.4 Gear Code and Soak Time Interdependence

[Table 3](#) shows how the validity of the soak time value depends upon the gear code that was reported. An **X** indicates that the field must have a provided value based upon the gear code.

**Table 3: Gear Code and Soak Time Interdependence**

GEAR CODE	GEAR NAME	GEAR CODE DEPENDENT MESH_SIZE	GEAR CODE DEPENDENT GEARQTY	GEAR CODE DEPENDENT GEARSIZE	GEAR CODE DEPENDENT NUMBER OF HAULS	Soak Time Evaluation (RULE ID 169)
CST	CASTNET	X	NA	X	X	<i>tow_soak_time</i> cannot exceed time at sea
DIP	"DIP NET, COMMON"	X	X	X	X	<i>tow_soak_time</i> cannot exceed time at sea
DIV	DIVING GEAR	NA	NA	NA	NA	<i>tow_soak_time</i> cannot exceed time at sea
DRM	"DREDGE, MUSSEL"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
DRC	"DREDGE, OCEAN QUAHOG/SURF CLAM"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
DRO	"DREDGE, OTHER"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
DRS	"DREDGE, SCALLOP,SEA"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
DSC	"DREDGE, SCALLOP-CHAIN MAT"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
DRU	"DREDGE, URCHIN"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
DTC	"DREDGE,SCALLOP,CHAIN MAT,MOD"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
DTS	"DREDGE,SCALLOP,TURTLE DEFLECT"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
FYK	FYKE NET	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
GND	"GILL NET, DRIFT,LARGE MESH"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea

GEAR CODE	GEAR NAME	GEAR CODE DEPENDENT MESH_SIZE	GEAR CODE DEPENDENT GEARQTY	GEAR CODE DEPENDENT GEARSIZE	GEAR CODE DEPENDENT NUMBER OF HAULS	Soak Time Evaluation (RULE ID 169)
GNT	"GILL NET, DRIFT, SMALL MESH"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
GNO	"GILL NET, OTHER"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
GNR	"GILL NET, RUNAROUND"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
GNS	"GILL NET, SINK"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
HND	HAND LINE/ROD & REEL	NA	X	NA	NA	<i>tow_soak_time</i> cannot exceed time at sea
RAK	HAND RAKE	NA	NA	NA	NA	<i>tow_soak_time</i> cannot exceed time at sea
HRP	HARPOON	NA	X	NA	Mandatory unless value reported for <i>species</i> is NC (no catch)	<i>tow_soak_time</i> cannot exceed time at sea
LLB	"LONGLINE, BOTTOM"	NA	X	X	X	<i>tow_soak_time</i> can exceed time at sea
LLP	"LONGLINE, PELAGIC"	NA	X	X	X	<i>tow_soak_time</i> can exceed time at sea
OTB	"OTTER TRAWL, BEAM"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
OTF	"OTTER TRAWL, BOTTOM, FISH"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
OTO	"OTTER TRAWL, BOTTOM, OTHER"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
OTC	"OTTER TRAWL, BOTTOM, SCALLOP"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
OTS	"OTTER TRAWL, BOTTOM, SHRIMP"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
OHS	"OTTER TRAWL, HADDOCK SEPARATOR"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea

GEAR CODE	GEAR NAME	GEAR CODE DEPENDENT MESH_SIZE	GEAR CODE DEPENDENT GEARQTY	GEAR CODE DEPENDENT GEARSIZE	GEAR CODE DEPENDENT NUMBER OF HAULS	Soak Time Evaluation (RULE ID 169)
OTM	"OTTER TRAWL, MIDWATER"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
OTR	"OTTER TRAWL, RUHLE"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
TTS	"OTTER TRAWL,BOTTOM,SHRIMP,TWIN"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
OTT	"OTTER TRAWL,BOTTOM,TWIN"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
OBP	"OTTER TRAWL,LGMESH BELLY PANEL"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
PTB	"PAIR TRAWL, BOTTOM"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
PTM	"PAIR TRAWL, MIDWATER"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
PTW	"POT, CONCH/WHELK"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
PTC	"POT, CRAB"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
PTE	"POT, EEL"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
PTF	"POT, FISH"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
PTH	"POT, HAG"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
PTL	"POT, LOBSTER"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
PTO	"POT, OTHER"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
PTS	"POT, SHRIMP"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
PTX	"POTS, MIXED"	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
SED	"SEINE, DANISH"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea



GEAR CODE	GEAR NAME	GEAR CODE DEPENDENT MESH_SIZE	GEAR CODE DEPENDENT GEARQTY	GEAR CODE DEPENDENT GEARSIZE	GEAR CODE DEPENDENT NUMBER OF HAULS	Soak Time Evaluation (RULE ID 169)
SEH	"SEINE, HAUL"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
PUR	"SEINE, PURSE"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
SES	"SEINE, SCOTTISH"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
STS	"SEINE, STOP"	X	X	X	X	<i>tow_soak_time</i> * <i>number_of_hauls</i> cannot exceed time at sea
TRP	TRAP	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea
TGS	"TROLL LINE, GREEN STICK"	NA	X	NA	NA	<i>tow_soak_time</i> cannot exceed time at sea
WEI	WEIR	X	X	X	X	<i>tow_soak_time</i> can exceed time at sea

[Jump to: Table of Contents](#)

## 4.5 Quality Control Audit Rules

[Table 4](#) provides a list of audit rules. The audit rules engine checks the input data against the rules and, if there is an error, returns to the user an error notification as well as a text description of the error.

Find the error text descriptions also in [Table 4](#) in the column **Rule Description**.

**Table 4: Quality Control Audit Rules**

LEVEL	RULE ID	RULE NAME	RULE DESCRIPTION	RULE DETAIL
TRIP	3	DATE LAND BEFORE DATE SAIL	Evaluates a trip record to determine if the land date occurs before the sail date.	date_land must be greater than date_sail
TRIP	4	DATE LAND AFTER TODAYS DATE	Evaluates a trip record to determine if the land date occurs more than one hours in the future.	date_land must not be greater than one hour past the submission datetime.
TRIP	5	DATE SAIL AFTER TODAYS DATE	Evaluates a trip record to determine if the sail date occurs in the future (after todays date - date trip reported).	date_sail must not be greater than the submission datetime.
TRIP	14	TRIP ID DOES NOT MATCH VESSEL PERMIT NUMBER	Evaluates a trip record to determine if the trip ID matches the vessel permit number reported.	The first six digits of trip_id must be identical to vessel_permit_num.
TRIP	23	TRIP MISSING SUBTRIP RECORDS	Evaluates a trip record to determine if it is missing required subtrip records. All trips where no effort reason (trip_activity_type) equals 0 1 requires a subtrip record. Trips with an trip activity type equal to 1,2,3,4,5,6,7,8 are not required to submit a subtrip record.	If no_effort_trip_reason is 0, a subtrip array must exist.
TRIP	24	TRIP MISSING CATCH DATA	Evaluates a trip record to determine if it is missing required catch records. If no effort reason (trip_activity_type) equals 0, a catch array must exist.	If no_effort_trip_reason is 0, a catch array must exist.
TRIP	38	INVALID NUMBER OF ANGLERS	Evaluates a trip record to determine if the value of number of anglers is not empty or zero.	Checks if number of anglers is provided.
TRIP	40	DUPLICATE OF EXISTING TRIP WITH SAME PERMIT AND DATE SAIL	Evaluates if a trip with the same permit number and date sail are already entered into the production CID system. Evaluates the unique constraint defined by DOC_PNUM_DSAIL unique index. On 24-AUG-15 added code to account for re-submitted trips - so only valid where there is a duplicate trip but they do not have the same trip IDs.	A submitted trip may not have the same date_sail and vessel_permit_num as a previously submitted trip which does not have the same trip_id. A resubmitted trip must use the same trip_id as the trip record it is meant to update.

LEVEL	RULE ID	RULE NAME	RULE DESCRIPTION	RULE DETAIL
TRIP	69	TRIP DATE SAIL OVERLAPS AN EXISTING TRIP	Evaluates a trip record to determine if the vessel permit number submitted a trip that overlaps (date_sail is between another trip's date sail and land dates) with an existing trip in GARFO's database.	For the given vessel_permit_num, the timespan described by date_sail and date_land must not in any way overlap a previously submitted trip which does not have the same trip_id. A resubmitted trip must use the same trip_id as the trip record it is meant to update.
TRIP	161	INVALID OPERATOR NAME LENGTH	Evaluates operator first and last name for acceptable length.	The combined length of operator_first_name and operator_last_name must not exceed 35 characters.
TRIP	168	MINIMUM TRIP LENGTH NOT MET	Compares date_sail and date_land to determine that the length of the trip is possible	date_sail and date_land must represent a span of time that exceeds 29 minutes.
TRIP	175	INVALID NUMBER OF ANGLERS		Evaluates a trip record to determine if the value of number of anglers is not too big.
TRIP	177	INVALID OPERATOR NAME	Evaluates a trip record to determine if the operator name contains unexpected characters.	Evaluates a trip record to determine if the operator name contains unexpected characters.
TRIP	182	TRIP CONTAINS SPLIT EFFORT	Evaluates a trip for split efforts which may be haul by haul or in errors	Looks for split effort trips which may be haul by haul submissions or possible errors
TRIP	215	Invalid Number of Crew	Evaluates a trip record to determine if the value of number of crew is not empty or zero on all trips besides a recreational trip.	Checks if number of crew is provided unless it is a recreational trip
TRIP	231	NOT A PRODUCTION USER	Evaluates a trip record to determine if it is a test submission.	If a test user , no submission allowed in production.
TRIP	232	Catch quantity not equal to sold quantity	Evaluates the quantity of each species for which kept > 0 to assure that it does not exceed the amount of that species for which a disposition is reported.	Checks if catch quantity is equal with sold quantity
TRIP	233	Sold quantity not equal to catch quantity	Evaluates the quantity of each species for which a disposition is reported to assure it does not exceed the amount of that species reported kept	Checks if sold quantity is equal with catch quantity
SUBTRIP	6	GEARCODE INVALID	Evaluates a subtrip record to determine if the gear code is a valid gear code.	gearcode must exist in the available list of gear.

LEVEL	RULE ID	RULE NAME	RULE DESCRIPTION	RULE DETAIL
SUBTRIP	7	AREA INVALID	Evaluates a subtrip record to determine if the area is a valid area.	chart_area must exist in the available list of areas.
SUBTRIP	8	GEAR REPORTED WITHOUT A SPECIES	Evaluates a subtrip record to determine if gear has been reported without reporting a species catch - except when gearcode = CAR which is exempt from this requirement OR a set only trip which will report gear and not species.	If no_effort_trip_reason =1, and gearcode is not 'CAR', species may not be NULL. 'NC' is the correct species code for an effort with no catch.
SUBTRIP	22	ORPHAN SUBTRIP RECORD	Evaluates a subtrip record to determine if the subtrip is missing a parent trip record.	A subtrip array must exist as the child of a trip array.
SUBTRIP	30	SET ONLY TRIP MISSING REQUIRED VALUES	Evaluates a set-only trip to determine if the subtrip has all the required values.	A set-only trip must have values for gearcode, mesh, gear quantity, gear size fishing depth or chart area and lat/long/or loran.
SUBTRIP	35	INVALID SOAK TIME FOR GEAR CODE	The soak time indicated for a gear code is not valid.	The soak time indicated for a gear code is not valid.
SUBTRIP	36	INVALID GEAR QUANTITY OR NUMBER OF HAULS	Effort gear quantity or number of hauls invalid for gear code.	Effort gear quantity or number of hauls invalid for gear code.
SUBTRIP	39	SOAK TIME EXCEEDS LENGTH OF TRIP	Evaluates gearcode, soak time and hauls to determine if effort exceeds the time spent at sea.	If gearcode is not (0), then date_sail and date_land must represent a span of time that is greater than or equal to the sum of all tow_soak_time values in the given trip or the total value of tow_soak_time multiplied by number_of_hauls for all efforts.
SUBTRIP	162	INVALID AREA LATITUDE LONGITUDE	Evaluates a subtrip record to determine if the latitude / longitude reported describes a point in an appropriate area	The geographic point represented by latitude_or_loran1 and longitude_or_loran2 must exist within waters off of the U.S. Eastern Seaboard or within the Gulf of Mexico.
SUBTRIP	166	NUMBER OF HAULS MISSING	Evaluates a subtrip record to determine that number_of_hauls has a value when the gear code does not exempt that field from being required.	If gearcode is not 'HND', 'DIV', 'RAK', or 'HRP', number_of_hauls must not be NULL.
SUBTRIP	176	INVALID FISHING DEPTH	Evaluates a subtrip record to determine depth isn't too big	Evaluates a subtrip record to determine that fishing_depth has a value that is not too big

LEVEL	RULE ID	RULE NAME	RULE DESCRIPTION	RULE DETAIL
SUBTRIP	180	MESH SIZE MISSING OR INVALID VALUE FOR GEAR TYPE	If mesh size is required, field cannot be 0 or null. if mesh size is not required, field can only be null or 0	If mesh size is required, field can not be 0 or null. if mesh size is not required, field can only be null or 0
SUBTRIP	181	GEAR SIZE MISSING OR INVALID VALUE FOR GEAR TYPE	If gear size is required, field cannot be 0 or null. if gear size is not required, field can only be null or 0.	If gear size is required, field can not be 0 or null. if gear size is not required, field can only be null or 0.
SALE	9	SPECIES CODE INVALID	Evaluates a catch record to determine if the species code is invalid.	species must exist in the available list of species.
CATCH	9	SPECIES CODE INVALID	Evaluates a catch record to determine if the species code is invalid.	species must exist in the available list of species.
CATCH	28	SPECIES CATCH WEIGHT MISSING IN KEPT AND DISCARDED	Evaluates a catch record to determine if the species reported is missing pounds kept and discarded. Requirement is that at least kept or discarded must report a value greater than zero.	If species is not either 'NC' or 'DCP', a value greater than zero must be entered in either or both of kept and discarded.
CATCH	72	NO CATCH REPORTED A DEALER	Evaluates a catch record to determine if a dealer was reported for a species code of NC - No Catch	For a given catch record reporting species = 'NC' to indicate no catch, dealer_permit_number must be NULL.
SALE	183	INVALID DEALER NUMBER	Evaluates if provided dealer number on date_sold is valid	Dealer number must be valid on date_sold
SALE	184	DATE SOLD MISSING	Evaluates if date sold is provided when fish are offloaded	Date sold must be provided when amount sold is bigger then 0 and dealer code is not in list which excludes from date sold needed.
SALE	229	INVALID PORT CODE	Evaluates port code to ensure it is valid	If a sales record indicates a value for sold that is greater than zero, the value for offloading_port must exist in the the available list of ports.
SALE	230	PORT CODE IS MISSING	Evaluates if port code is valid	If a sales record indicates a value for sold that is greater than zero, offloading_port must not be NULL.
SALE	234	DATE SOLD BEFORE DATE LAND	Evaluates the date sold to determine if it occurs before the date land.	date_sold must be greater than date_land

## 4.6 Quality Control Audit Rules for Lobster and Crab

[Table 5](#) provides the audit rules that pertain to lobster and crab trips. (At this publishing date, Rule IDs have not been finalized.)

**Table 5: Audit Rules (Lobster and Crab)**

LEVEL	RULE ID	RULE NAME	RULE DESCRIPTION	RULE DETAIL
TRIP	To Be Determined	NUMBER OF POTS IN AREA NOT EXCEED TOTAL IN WATER	Evaluates a subtrip (effort) record to determine that the number of pots in the statistical area do not exceed the total number of Lobster/Crab pots in water	Number of Pots in This Area should not exceed Total Lobster/Crab Pots in Water
TRIP	To Be Determined	NUMBER OF BUOY LINES IN AREA NOT EXCEED TOTAL IN WATER	Evaluates a subtrip (effort) record to determine that the number of buoy lines in the statistical area do not exceed the total number of buoy lines in the water	Number of Buoy Lines in This Area should not exceed Total Lobster/Crab Buoy Lines in Water
TRIP	To Be Determined	TOTAL BUOY LINES NOT EXCEED TOTAL POTS IN WATER	Evaluates a trip to determine that the total number of Lobster/Crab buoy lines in the water do not exceed the total number of Lobster/Crab pots in the water	Total Lobster/Crab Buoy Lines in Water should not exceed the Total Lobster/Crab Pots in Water
TRIP	To Be Determined	NOT APPLICABLE	Ensures the number of Crew is not null	Number of Crew cannot be nullable
TRIP	To Be Determined	NOT APPLICABLE	Ensures the total number of Lobster/Crab Pots in the water is not null	Total Lobster/Crab Pots in Water cannot be nullable
TRIP	To Be Determined	NOT APPLICABLE	Ensures the average Lobster/Crab pots per string in the water is not null	Average Lobster/Crab Pots Per String in Water cannot be nullable
TRIP	To Be Determined	NOT APPLICABLE	Ensures the total number of Lobster/Crab buoy lines in the water is not null	Total Lobster/Crab Buoy Lines in Water cannot be nullable
SUBTRIP	To Be Determined	BUOY LINES IN AREA DON'T EXCEED POTS IN AREA	Evaluates a subtrip (effort) record to determine that if the reported gear is lobster or crab pots, the number of buoy lines in the statistical area do not exceed the number of pots in this statistical area	If gear IN ('PTL' , 'PTLL', 'PTC', 'PTCL') and Number of Buoy Lines in This Area should not exceed the Number of Pots in This Area
SUBTRIP	To Be Determined	STRINGS HAULED IN AREA MAY NOT BE NULL	Evaluates a subtrip (effort) record to determine that if the reported gear is lobster or crab pots, the number of strings hauled in the statistical area is not null	If gear IN ('PTL' , 'PTLL', 'PTC', 'PTCL') and Number of Strings Hauled in This Area cannot be nullable

LEVEL	RULE ID	RULE NAME	RULE DESCRIPTION	RULE DETAIL
SUBTRIP	To Be Determined	SOAK TIME CAN NOT BE NULL	Evaluates a subtrip (effort) record to determine that if the reported gear is lobster or crab pots, the average soak time of the strings hauled in the statistical area is not null	If gear IN ('PTL' , 'PTLL', 'PTC', 'PTCL') and Average Soak Time of Strings Hauled This Area cannot be nullable
SUBTRIP	To Be Determined	POTS IN AREA MAY NOT BE NULL	Evaluates a subtrip (effort) record to determine that if the reported gear is lobster or crab pots, the number of pots in the statistical area is not null	If gear IN ('PTL' , 'PTLL', 'PTC', 'PTCL') and Number of Pots in This Area cannot be nullable
SUBTRIP	To Be Determined	POTS HAULED MAY NOT BE NULL	Evaluates a subtrip (effort) record to determine that if the reported gear is lobster or crab pots, the number of pots hauled in the statistical area is not null	If gear IN ('PTL' , 'PTLL', 'PTC', 'PTCL') and Number of Pots Hauled in This Area cannot be nullable
SUBTRIP	To Be Determined	BUOY LINES IN AREA MAY NOT BE NULL	Evaluates a subtrip (effort) record to determine that if the reported gear is lobster or crab pots, the number of buoy lines in the statistical area is not null	If gear IN ('PTL' , 'PTLL', 'PTC', 'PTCL') and Number of Buoy Lines in This Area cannot be nullable
SUBTRIP	To Be Determined	AVERAGE POTS PER STRING MAY NOT BE NULL	Evaluates a subtrip (effort) record to determine that if the reported gear is lobster or crab pots, the average number of pots per strings hauled in the statistical area is not null.	If gear IN ('PTL' , 'PTLL', 'PTC', 'PTCL') and Average Number of Pots Per Strings Hauled This Area cannot be nullable
TRIP	To Be Determined	LOBSTER FORM REQUIRED FOR TRIPS STARTING APRIL 1, 2024	Requires the use of the Lobster reporting form for trips starting on April 1, 2024 and after	Gear IN ('PTL' , 'PTLL') and DATE_SAIL >= April 1, 2024
TRIP	To Be Determined	JONAH CRAB FORM REQUIRED FOR TRIPS STARTING APRIL 1, 2024	Requires the use of the jonah crab reporting form for trips starting on April 1, 2024 and after	Gear IN ('PTC', 'PTCL') and DATE_SAIL >= April 1, 2024 and species = 'CRJ'

## 4.7 Regulatory References

In July 2011, the Greater Atlantic Regional Fisheries Office (GARFO) of the National Marine Fisheries Service (NMFS) approved the use of electronic reporting for the completion of Vessel Trip Reports (VTRs) on a limited, voluntary basis. The Federal Register, in the section indicated, outlines the regulatory requirements that eVTR software must meet:

50 CFR 648.7(b)(1)(i)

Use the following hypertext link to jump to the appropriate section:

[Link to the Federal Register language about eVTR software requirements](#)

## 5 Requirements for Lobster Reporting (Report\_Type=2)

If your organization plans to offer reporting for lobster trips, this section provides the information necessary for programmers to implement the business rules and data elements required of eVTR applications for use by the lobster fleet.

### 5.1 Report Type Field

Add the field `report_type` to the top level of the JSON object.

- Values for `report_type` may be:  

```
[{ "VALUE": "OTHER", "KEY": 1}  
, { "VALUE": "LOBSTER", "KEY": 2}]
```
- Datatype: INTEGER
- Format: [1-3]

### 5.2 Trip Type Label

This section contains information about changes to the Trip Type label.

#### 5.2.1 New Trip Type Values

The GUI will continue to display the label `Trip Type`, with three additional options in addition to the four existing trip types.

```
Clam ITQ - Commercial  
Commercial - Lobster/Jonah Crab  
Lobster Charter
```

#### 5.2.2 Existing Trip Type Values (For Reference)

The existing `trip_type` values, which will not change, are as follows:

```
[{"VALUE": "Commercial", "KEY": 1}  
, {"VALUE": "Party", "KEY": 2}  
, {"VALUE": "Charter", "KEY": 3}  
, {"VALUE": "Private Recreational-Private use of a federally permitted vessel",  
"KEY": 6}]
```

#### 5.2.3 Trip Type Label Examples

The user's selection under the **Trip Type** label determines how the `report_type` and `trip_type` fields are populated.

##### Examples

- User selects **Lobster Charter** — produces `trip_type=3`, `report_type=2`
- Current state VTR: User selects **Commercial** — produces `trip_type=1`, `report_type=1`

[Jump to: Table of Contents](#)



## 5.3 *New Trip-level Input Fields for Lobster (Report\_Type=2)*

This section provides the new trip-level input fields for `report_type = 2` (Lobster)

### 5.3.1 End of a Trip

#### 5.3.1.1 *Total Number of Buoy Lines All Chart Areas*

- JSON Field: `lines_in_water_total`
- Trip Reports Attribute: `BUOY LINES IN WATER COUNT`
- Label: Total Lobster/Crab Buoy Lines In Water

New at February 2024 release

- Datatype: `INTEGER`
- Format: 999

#### 5.3.1.2 *Average Number of Pots per string All Chart Areas*

- JSON Field: `avg_traps_per_string`
- Trip Reports Attribute: `PER_STRING_AVERAGE`
- Label: Average Lobster/Crab Pots per string
- Datatype: `NUMBER`
- Format: 99.99

#### 5.3.1.3 *Total Number of Strings Hauled (trip)*

- JSON Field: `strings_hauled_per_trip`
- Trip Reports Attribute: `HAUL_STRING_COUNT`
- Label: Number Of Strings Hauled This Trip
- Datatype: `INTEGER`
- Format: 999

#### 5.3.1.4 *Total Pots All Chart Areas*

- JSON Field: `traps_in_water_total`
- Trip Reports Attribute: `IN_WATER_COUNT`
- Label: Total Lobster/Crab Pots in Water
- Datatype: `integer`
- Format: 9999

#### 5.3.1.5 *Average Soak Time (trip)*

- JSON Field: `avg_soak_time_per_trip`
- Trip Reports Attribute: `SOAK_DURATION_AVERAGE`
- Label: Average Soak Time of Strings Hauled This Trip
- Datatype: `integer`
- Format: 999.99
- Units: Hours

### **5.3.1.6 Unit of Measurement Average Soak Time (trip)**

- JSON Field: `avg_soak_time_per_trip_uom`
- Trip Reports Attribute: NOT MAPPED
- Datatype: `STRING`
- Value: Hours

### **5.3.1.7 Display Order for the Trip-level Input Fields**

Display the new trip-level input fields in the following order:

- 1) Total Lobster/Crab Pots in Water
- 2) Average Lobster/Crab Pots Per String in Water
- 3) Total Lobster/Crab Buoy Lines in Water
- 4) Number of Strings Hauled This Trip
- 5) Average Soak Time of Strings Hauled This Trip

[Jump to: Table of Contents](#)

## **5.4 Effort-level Updates for Lobster (Report\_Type=2)**

This section provides the new effort-level input fields for `report_type = 2` (Lobster).

### **5.4.1 New Effort-level Input Fields Entered for Each Effort**

#### **5.4.1.1 Number of Traps Hauled in Each NMFS Statistical Area Fished**

- JSON Field: `traps_per_effort`
- Trip Reports Attribute: `HAUL_GEAR_COUNT`
- Label: Number of Pots Hauled This Area
- Datatype: `INTEGER`
- Format: 9999

#### **5.4.1.2 Number of Traps in the Water in Each NMFS Statistical Area Fished**

- JSON Field: `traps_in_water_per_area`
- Trip Reports Attribute: `IN_WATER_COUNT`
- Label: Number of Pots in This Area
- Datatype: `INTEGER`
- Format: 9999

### **5.4.1.3 Average Number of Traps per String in Each NMFS Statistical Area Fished**

- JSON Field: `traps_per_string_per_area`
- Trip Reports Attribute: `PER_STRING_AVERAGE`
- Label: Average number of traps per string in this Stat area
- Datatype: `NUMBER`
- Format: `99.99`

### **5.4.1.4 Number of Buoy Lines in Each NMFS Statistical Area Fished**

- JSON Field: `lines_in_water_per_area`
- Trip Reports Attribute: `BUOY_LINES_IN_WATER_COUNT`
  - Newly added in February 2024
- Label: Number of Buoy Lines in this Area
- Datatype: `INTEGER`
- Format: `999`

### **5.4.1.5 Total Number of Strings Hauled (Effort)**

- JSON Field: `strings_hauled_per_effort`
- Trip Reports Attribute: `HAUL_STRING_COUNT`
- Label: Number of Strings Hauled This Area
- Datatype: `INTEGER`
- Format: `999`

### **5.4.1.6 Average Soak Time (Effort)**

- 1. JSON Field: `avg_soak_time_per_effort`
- 2. Trip Reports Attribute: `SOAK_DURATION_AVERAGE`
- 3. Label: Average Soak Time of Strings Hauled This Area
- 4. Datatype: `NUMBER`
- 5. Format: `999.99`
- 6. Units: `Hours`

### 5.4.1.7 Unit of Measurement Average Soak Time (Effort)

- 1. JSON Field: `avg_soak_time_per_effort_uom`
- 2. Trip Reports Attribute: NOT MAPPED
- 3. Datatype: `STRING`
- 4. Value: `Hours`

### 5.4.1.8 Display Order for the Effort-Level Input Fields

Order the Effort-level input fields as follows:

- 1) Number of Pots in This Area
- 2) Number of Buoy Lines in This Area
- 3) Number of Pots Hauled in This Area
- 4) Number of Strings Hauled in This Area
- 5) Average Number of Pots Per Strings Hauled This Area
- 6) Average Soak Time of Strings Hauled This Area

[Jump to: Table of Contents](#)

## 5.5 Existing Effort-level Fields for Lobster (Report\_Type=2)

The following Effort-level input fields remain unchanged:

- `mesh_size`

[Jump to: Table of Contents](#)

## 5.6 Removed Effort-level Fields for Lobster (Report\_Type=2)

For `report_type = 2` (Lobster), there are deprecated Effort-level input fields to account for. You must **Remove** the following fields for `report_type = 2` (Lobster):

- Gear Size (`gearsizes`)
- Gear Quantity (`gearqty`)
- Number of Hauls (`number_of_hauls`)
- Tow/Soak Time (`tow_soak_time`)

---

**NOTE:** Removed fields may be either omitted or left empty.

---

[Jump to: Table of Contents](#)

## 5.7 Gear Options for Lobster (Report\_Type=2)

The following conditions apply for implementing gear options for lobster trips.

- All gears are available for selection on a lobster trip.
- Not all gears are legal to fish in correlation with trap gear, but several types of non-trap gear are permissible on lobster trips. GARFO allows a captain on a lobster trip to report any gear type.
- Upon enactment of the lobster rule, GARFO must exclude pot gear from regular commercial trips. In the interim period, captains may report with the new form, but will not be required to do so. After April 1, 2024, it will not be allowed to report lobster trips using trap gear on the regular VTR.

## 5.8 Endpoint

---

**WARNING: DO NOT submit test data to the production endpoint!**

---

- Production: [https://apps-garfo.fisheries.noaa.gov/base\\_api/evtr/v\\_1\\_lobster](https://apps-garfo.fisheries.noaa.gov/base_api/evtr/v_1_lobster)
- Test: [https://apps-garfo.fisheries.noaa.gov/base\\_api/evtr/v\\_1\\_lobster/test](https://apps-garfo.fisheries.noaa.gov/base_api/evtr/v_1_lobster/test)

## 5.9 Example JSONs

This section provides a JSON example for several possible types of Effort on a Lobster trip (report\_type=2).

### 5.9.1 Example Lobster No Effort JSON

```
{
  "client_id": "*****",
  "token": "*****",
  "test_user": true,
  "vendor_id": "37901",
  "vendor_app_name": "Fish Online React",
  "vendor_software_version": "13.1.1",
  "vendor_app_platform": "MacIntel",
  "record_type": "1",
  "trip_count": 1,
  "vendor_message_id": null,
  "trip_id": "55555524022105",
  "vessel_permit_num": "555555",
  "vessel_registration": "TEST1111",
  "date_sail": "2402210500",
  "date_land": "2402222235",
  "report_type": "2",
  "trip_type": "1",
  "number_of_crew": "6",
  "number_of_anglers": "",
  "operator_first_name": "Test",
  "operator_last_name": "User",
  "operator_ext_name": "",
  "operator_permit_number": "99938094",
  "operator_password": null,
```

```
"no_effort_trip_reason": "1",
"trip_comments": "",
"traps_in_water_total": "",
"avg_traps_per_string": "",
"lines_in_water_total": ""
}
```

[Jump to: Table of Contents](#)

## 5.9.2 Example Lobster Only Effort JSON

```
{
  "client_id": "*****",
  "token": "*****",
  "test_user": true,
  "vendor_id": "12345",
  "vendor_app_name": "Your App Name Here",
  "vendor_software_version": "1.2.3",
  "vendor_app_platform": "MacIntel",
  "record_type": "1",
  "trip_count": 1,
  "vendor_message_id": null,
  "trip_id": "55555524030514",
  "vessel_permit_num": "555555",
  "vessel_registration": "TEST1111",
  "date_sail": "2402050612",
  "date_land": "2402062313",
  "report_type": "2",
  "trip_type": "1",
  "number_of_crew": "5",
  "number_of_anglers": "",
  "operator_first_name": "Test",
  "operator_last_name": "User",
  "operator_ext_name": "",
  "operator_permit_number": "99938094",
  "operator_password": null,
  "no_effort_trip_reason": "0",
  "trip_comments": "",
  "traps_in_water_total": "3",
  "avg_traps_per_string": "33",
  "lines_in_water_total": "3",
  "subtrips": [
    {
      "record_type": "1",
      "gearcode": "PTLL",
      "gearqty": "",
      "gearsizes": "",
      "mesh_size": "5",

```

```

    "latitude_or_loran1": "41",
    "longitude_or_loran2": "72",
    "chart_area": "500",
    "fishing_depth": "50",
    "fishing_depth_uom": "fa",
    "number_of_hauls": "",
    "tow_soak_time": "",
    "datetime_haul_start": null,
    "datetime_haul_end": null,
    "serial_num": "",
    "catch": [
      {
        "record_type": "1",
        "species": "LOB",
        "kept": "150",
        "discarded": "0",
        "uom": "LBS"
      }
    ],
    "traps_per_effort": "2",
    "traps_in_water_per_area": "5",
    "traps_per_string_per_area": "3",
    "lines_in_water_per_area": "10",
    "strings_hauled_per_effort": "4",
    "avg_soak_time_per_effort": "6",
    "avg_soak_time_per_effort_uom": "hours"
  }
],
"effort_record_count_by_trip": 1,
"effort_count": 1,
"catch_count": 1,
"sales": [
  {
    "species": "LOB",
    "dealer_permit_number": "1234",
    "sold": "150",
    "date_sold": "240207",
    "offloading_port": "220999",
    "uom": "LBS",
    "dealer_name": "SOME FISH COMPANY",
    "price_per_bushel": "",
    "allocation_numbers": [],
    "tag_numbers": []
  }
]
}

```

[Jump to: Table of Contents](#)

### 5.9.3 Example Lobster Effort with Bycatch JSON

```
{
  "client_id": "*****",
  "token": "*****",
  "test_user": true,
  "vendor_id": "12345",
  "vendor_app_name": "Your App Name Here",
  "vendor_software_version": "1.2.3",
  "vendor_app_platform": "MacIntel",
  "record_type": "1",
  "trip_count": 1,
  "vendor_message_id": null,
  "trip_id": "41041024020906",
  "vessel_permit_num": "410410",
  "vessel_registration": "910012",
  "date_sail": "2402090625",
  "date_land": "2402101038",
  "report_type": "2",
  "trip_type": "1",
  "number_of_crew": "6",
  "number_of_anglers": "",
  "operator_first_name": "Test",
  "operator_last_name": "User",
  "operator_ext_name": "",
  "operator_permit_number": "99938094",
  "operator_password": null,
  "no_effort_trip_reason": "0",
  "trip_comments": "",
  "traps_in_water_total": "10",
  "avg_traps_per_string": "7",
  "lines_in_water_total": "8",
  "subtrips": [
    {
      "record_type": "1",
      "gearcode": "PTLL",
      "gearqty": "",
      "gearsizes": "",
      "mesh_size": "5",
      "latitude_or_loran1": "41:25",
      "longitude_or_loran2": "71:10",
      "chart_area": "500",
      "fishing_depth": "50",
      "fishing_depth_uom": "fa",
      "number_of_hauls": "",
      "tow_soak_time": "",
      "datetime_haul_start": null,
      "datetime_haul_end": null,
    }
  ]
}
```



```
"serial_num": "",
"catch": [
  {
    "record_type": "1",
    "species": "LOB",
    "kept": "75",
    "discarded": "3",
    "uom": "LBS"
  },
  {
    "record_type": "1",
    "species": "CRB",
    "kept": "45",
    "discarded": "0",
    "uom": "LBS"
  }
],
"traps_per_effort": "6",
"traps_in_water_per_area": "9",
"traps_per_string_per_area": "2",
"lines_in_water_per_area": "18",
"strings_hauled_per_effort": "12",
"avg_soak_time_per_effort": "4",
"avg_soak_time_per_effort_uom": "hours"
}
],
"effort_record_count_by_trip": 1,
"effort_count": 1,
"catch_count": 2,
"sales": [
  {
    "species": "LOB",
    "dealer_permit_number": "1639",
    "sold": "75",
    "date_sold": "240211",
    "offloading_port": "240116",
    "uom": "LBS",
    "dealer_name": "COMMERCIAL LOBSTER CO INC",
    "price_per_bushel": "",
    "allocation_numbers": [],
    "tag_numbers": []
  },
  {
    "species": "CRB",
    "dealer_permit_number": "1234",
    "sold": "45",
    "date_sold": "240211",
    "offloading_port": "240116",
```

```

        "uom": "LBS",
        "dealer_name": "SOME FISH COMPANY",
        "price_per_bushel": "",
        "allocation_numbers": [],
        "tag_numbers": []
    }
]
}

```

[Jump to: Table of Contents](#)

#### 5.9.4 Example Lobster Multiple Efforts JSON

```

{
  "client_id": "*****",
  "token": "*****",
  "test_user": true,
  "vendor_id": "12345",
  "vendor_app_name": "Your App Name Here",
  "vendor_software_version": "1.2.3",
  "vendor_app_platform": "MacIntel",
  "record_type": "1",
  "trip_count": 1,
  "vendor_message_id": null,
  "trip_id": "41041024021207",
  "vessel_permit_num": "410410",
  "vessel_registration": "910012",
  "date_sail": "2402120715",
  "date_land": "2402132217",
  "report_type": "2",
  "trip_type": "1",
  "number_of_crew": "9",
  "number_of_anglers": "",
  "operator_first_name": "Test",
  "operator_last_name": "User",
  "operator_ext_name": "",
  "operator_permit_number": "99938094",
  "operator_password": null,
  "no_effort_trip_reason": "0",
  "trip_comments": "",
  "traps_in_water_total": "6",
  "avg_traps_per_string": "3",
  "lines_in_water_total": "12",
  "subtrips": [
    {
      "record_type": "1",
      "gearcode": "PTLL",
      "gearqty": "",

```

```
"gearsizesize": "",
"mesh_size": "4",
"latitude_or_loran1": "41",
"longitude_or_loran2": "72",
"chart_area": "500",
"fishing_depth": "50",
"fishing_depth_uom": "fa",
"number_of_hauls": "",
"tow_soak_time": "",
"datetime_haul_start": null,
"datetime_haul_end": null,
"serial_num": "",
"catch": [
  {
    "record_type": "1",
    "species": "LOB",
    "kept": "50",
    "discarded": "2",
    "uom": "LBS"
  }
],
"traps_per_effort": "9",
"traps_in_water_per_area": "5",
"traps_per_string_per_area": "4",
"lines_in_water_per_area": "7",
"strings_hauled_per_effort": "2",
"avg_soak_time_per_effort": "6",
"avg_soak_time_per_effort_uom": "hours"
},
{
  "record_type": "1",
  "gearcode": "HND",
  "gearqty": "3",
  "gearsizesize": "",
  "mesh_size": "",
  "latitude_or_loran1": "41:50",
  "longitude_or_loran2": "72:25",
  "chart_area": "500",
  "fishing_depth": "20",
  "fishing_depth_uom": "fa",
  "number_of_hauls": "",
  "tow_soak_time": "01:35",
  "datetime_haul_start": null,
  "datetime_haul_end": null,
  "serial_num": "",
  "catch": [
    {
      "record_type": "1",
```

```

        "species": "ALB",
        "kept": "65",
        "discarded": "0",
        "uom": "LBS"
    }
],
    "traps_per_effort": "",
    "traps_in_water_per_area": "",
    "traps_per_string_per_area": "",
    "lines_in_water_per_area": "",
    "strings_hauled_per_effort": "",
    "avg_soak_time_per_effort": "",
    "avg_soak_time_per_effort_uom": "hours"
}
],
"effort_record_count_by_trip": 2,
"effort_count": 2,
"catch_count": 2,
"sales": [
    {
        "species": "LOB",
        "dealer_permit_number": "1234",
        "sold": "25",
        "date_sold": "240214",
        "offloading_port": "220999",
        "uom": "LBS",
        "dealer_name": "SOME FISH COMPANY",
        "price_per_bushel": "",
        "allocation_numbers": [],
        "tag_numbers": []
    },
    {
        "species": "LOB",
        "dealer_permit_number": "1235",
        "sold": "25",
        "date_sold": "240215",
        "offloading_port": "240115",
        "uom": "LBS",
        "dealer_name": "OTHER FISH COMPANY",
        "price_per_bushel": "",
        "allocation_numbers": [],
        "tag_numbers": []
    },
    {
        "species": "ALB",
        "dealer_permit_number": "1236",
        "sold": "65",
        "date_sold": "240215",

```

```

        "offloading_port": "240115",
        "uom": "LBS",
        "dealer_name": "SWIMMING FISH COMPANY",
        "price_per_bushel": "",
        "allocation_numbers": [],
        "tag_numbers": []
    }
]
}

```

[Jump to: Table of Contents](#)

## 6 Requirements for Clam Reporting (Report\_Type=3)

### 6.1 Report Type Field

GARFO has added the field `report_type` to the top level of the JSON object.

Values for `report_type` may be:

```

[ { "VALUE": "OTHER", "KEY": 1 }
, { "VALUE": "LOBSTER", "KEY": 2 }
, { "VALUE": "CLAM", "KEY": 3 } ]

```

The GUI will continue to display the label **Trip Type**, with three additional options, Clam ITQ, Lobster Commercial Pot Gear, and Lobster Charter Pot Gear in addition to the four current trip types.

For reference, the existing `trip_type` values, which will not change, are

```

[ { "VALUE": "Commercial", "KEY": 1 }
, { "VALUE": "Party", "KEY": 2 }
, { "VALUE": "Charter", "KEY": 3 }
, { "VALUE": "Private Recreational-Private use of a federally permitted vessel",
"KEY": 6 } ]

```

### 6.1.1 GUI Examples

The examples in this section show how Fish Online's iOS and REACT apps format the **Trip Type** fields for display.

Figure 2: iOS Select Trip Type (GUI)

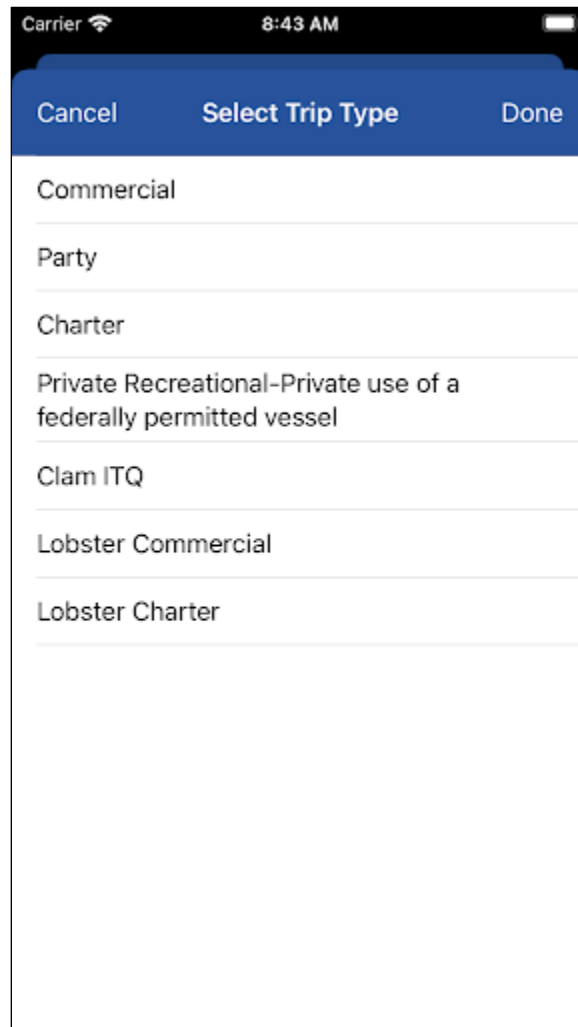
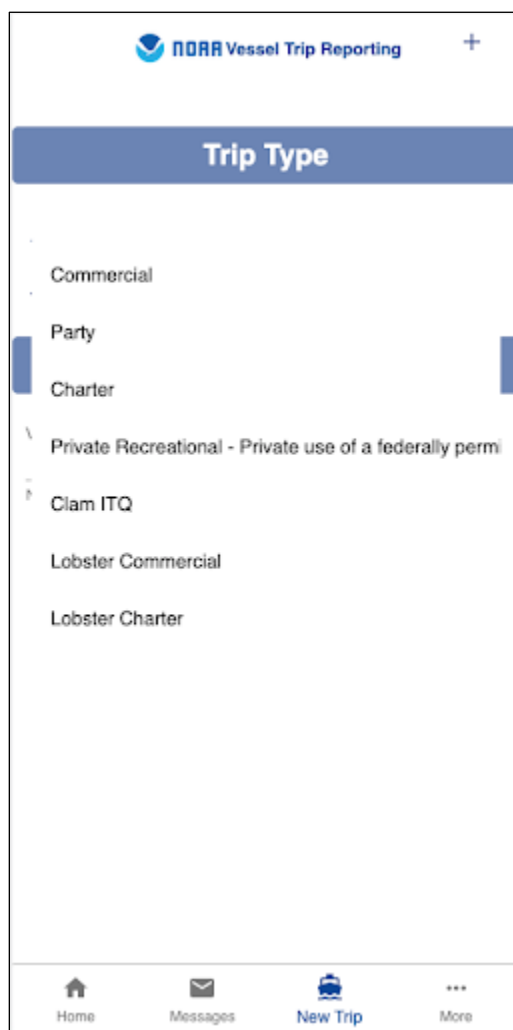


Figure 3: REACT Select Trip Type (GUI)



The user's selection under the **Trip Type** label determines how the `report_type` and `trip_type` fields are populated.

### Examples

- User selects **Lobster Charter** —produces `trip_type=3`, `report_type=2`
- User selects **Clam ITQ** — produces `trip_type=1`, `report_type=3`
- Current state VTR: User selects **Commercial** — produces `trip_type=1`, `report_type=1`

## 6.2 Catch-level Changes for Clams (Report\_Type=3)

This section provides information about the changes required to Catch-level input when `report_type=3`.

### 6.2.1 Enter Quantity Field

Placeholder text changes to the **Enter Quantity** Field when `report_type=3`.

- Species JSON must contain `x_rpt_type` key for species to be excluded for a given report type.
  - Remove `hms_species` key.
  - Hide specific species codes when `report_type=3`. Refer to [Table 6](#) for these codes.

Table 6: Species Codes to Hide When Report Type = 3

Species Code	Species Name	Key
CLA	CLAM, ARCTIC SURF	80983
CLH	HARD QUAHOG	81496
CLQBG	CLAM,OCEAN QUAHOG/BAG	81343
CLQU	OCEAN QUAHOG	81343-2
CLSO	CLAM, SOFT	81692
CLSU	CLAM, SURF	80944

## 6.2.2 Unit of Measurement (UOM) = Bushels

- Trigger is CLHB, CLQUB, CLSOB, CLSUB target species code in **Catch**.

## 6.2.3 New Gear Options

For Clam trips, the following are new **Gear** options:

- DRC
- RAK

## 6.3 New Disposition Input Fields for Clams (Report Type = 3)

For report\_type = 3, prompt for the following when a disposition includes SCOQ target species (CLHB, CLQUB, CLSOB, CLSUB).

### 6.3.1 Tag Number / Tag Range

- JSON Field: tag\_numbers []
- Label:

### 6.3.2 Allocation Number(s)

- JSON Field: allocation\_numbers []
- Label:

### 6.3.3 Price per Bushel

- JSON Field: price\_per\_bushel
- Label:

---

**NOTE:** Only ask these questions for: CLHB, CLQUB, CLSOB, CLSUB

---

[Jump to: Table of Contents](#)



## 6.4 JSON Example for Report\_Type=3

```
{
  "client_id": "*****",
  "token": "*****",
  "test_user": true,
  "vendor_id": "12345",
  "vendor_app_name": "Your App Name Here",
  "vendor_software_version": "1.2.3",
  "vendor_app_platform": "MacIntel",
  "record_type": "1",
  "trip_count": 1,
  "vendor_message_id": null,
  "trip_id": "55555524022706",
  "vessel_permit_num": "555555",
  "vessel_registration": "TEST1111",
  "date_sail": "2402270600",
  "date_land": "2403132355",
  "report_type": "3",
  "trip_type": "1",
  "number_of_crew": "8",
  "number_of_anglers": "",
  "operator_first_name": "Test",
  "operator_last_name": "User",
  "operator_ext_name": "",
  "operator_permit_number": "99938094",
  "operator_password": null,
  "no_effort_trip_reason": "0",
  "trip_comments": "",
  "subtrips": [
    {
      "record_type": "1",
      "gearcode": "DRC",
      "gearqty": "2",
      "gearsizes": "47",
      "mesh_size": "3",
      "latitude_or_loran1": "41",
      "longitude_or_loran2": "72",
      "chart_area": "500",
      "fishing_depth": "50",
      "fishing_depth_uom": "fa",
      "number_of_hauls": "50",
      "tow_soak_time": "01:35",
      "datetime_haul_start": null,
      "datetime_haul_end": null,
      "serial_num": "",
      "catch": [

```

```
        "record_type": "1",
        "species": "CRBB",
        "kept": "50",
        "discarded": "0",
        "uom": "BU"
      }
    ]
  }
],
"effort_record_count_by_trip": 1,
"effort_count": 1,
"catch_count": 1,
"sales": [
  {
    "species": "CRBB",
    "dealer_permit_number": "1234",
    "sold": "50",
    "date_sold": "240313",
    "offloading_port": "240115",
    "uom": "BU",
    "dealer_name": "SOME FISH COMPANY",
    "price_per_bushel": "",
    "allocation_numbers": [],
    "tag_numbers": []
  }
]
```

[Jump to: Table of Contents](#)

## 7 Document Revision History

Read this section to obtain revision history information for this guide.

### 7.1 Revision 1.0 Updates

Release Date: November 12, 2019

- No revisions (initial release)

### 7.2 Revision 1.1 Updates

Release Date: December 16, 2019

- **Section 4.5: Quality Control Audit Rules**
  - Removed the following audit rules from the Quality Control Audit Rules table:  
Rule IDs 10, 11, 12, 13, 17, 29, 31, 66, 67, 70, 71, 124, 125, 126, 127, 128, 129, 153, 154, 163, 164, 165, 170, 173, 174, 175, 176.  
The above audit rules have been removed from use by eVTR Software Development.
  - Added a column to the Quality Control Audit Rules table.  
The new column, *Audit Details*, provides additional information about conditions that would violate each referenced audit rule.

### 7.3 Revision 1.2 Updates

Release Date: April 20, 2020

- **Section 3.2.3: Returns From Authentication Request**
  - Converted the following fields to lowercase: *token*, *client\_id*, *operator\_key*, *vtr\_preferences*  
Code is case-sensitive; fields must be lowercase. (Exception is *Status*)
- **Section 3.3.3: JSON Format: Catch and Sales Data Together (Test)**
  - Added *client\_id* to JSON code example.  
The *client\_id* is returned with a correct authorization request, and the value needs to be submitted with the eVTR.
- **Section 3.3.4: JSON Format: Separate Arrays for Catch and Sales Data (Test)**
  - Added *client\_id* to JSON code example.  
The *client\_id* is returned with a correct authorization request, and the value needs to be submitted with the eVTR.
  - Removed *operator\_password*.  
Deprecated.
- **Section 4.2: eVTR Form Specification**
  - Added *client\_id* entry to specification.  
The *client\_id* is returned with a correct authorization request, and the value needs to be submitted with the eVTR.

## 7.4 Revision 1.2.1 Updates

Release Date: July 20, 2020

- Section 4.3: **Mandatory Use of a Field in eVTR Based on Trip Activity Type**
  - Expanded *number\_of\_anglers* specification
  - Added Trip Type 6 (private/recreational)

## 7.5 Revision 1.3 Updates

Release Date: January 15, 2021

- **Global edit:** Changed value **No Data** to value **NA** throughout the guide.  
**Context:** **NA** indicates that there is no value or instruction from GARFO concerning that particular instance.  
**Affected sections in the guide:**
  - Section 2.4: **File Format**
  - Section 4: **eVTR Reference**
  - Section 4.2: **eVTR Form Specification**
  - Section 4.3: **Mandatory Use of a Field in eVTR Based on Trip Activity Type**
  - Section 4.4: **Gear Code and Soak Time Interdependence**
- **Global edit:** Changed base URL to [https://apps-garfo.fisheries.noaa.gov/base\\_api/](https://apps-garfo.fisheries.noaa.gov/base_api/) throughout the guide.  
**Context:** This URL forms the basis of all requests to the eVTR API module. Create specific requests by appending request strings on this URL.  
**Affected sections in the guide:** Section 3.1.1 forward through the API discussion.
- Section 3.2.2: **Authentication**
  - Added the following fields: *vendor\_id*, *vendor\_software\_version*, *vendor\_app\_name*, *vendor\_app\_platform*  
Values that more precisely identify the app.
- Section 3.2.2.1: **App Platform**
  - New section  
More information about the *vendor\_app\_platform* field.
- Section 3.2.3.1: **Return From Correct Authorization**
  - Added the following fields: *token\_expires*, *entitlements*.
    - *token\_expires*  
Displays date/time expiration of authorization token.
    - *entitlements*  
Displays vessel(s) that the user is entitled to submit eVTRs for.
- Section 3.3.3.2: **Description**
  - Added the following fields: *vendor\_app\_name*, *vendor\_app\_platform*  
Fields to provide more vendor information.
  - Removed the following fields: *trip\_count*, *effort\_count*, *catch count*, *effort\_record\_count\_by\_trip*, *catch\_record\_count\_by\_subtrip*, *species\_count*.  
Deprecated.

- Section 3.3.4.2: **Description**
  - Added the following fields: *vendor\_app\_name*, *vendor\_app\_platform*  
Fields to provide more vendor information.
  - Removed the following fields: *trip\_count*, *effort\_count*, *catch count*, *effort\_record\_count\_by\_trip*, *catch\_record\_count\_by\_subtrip*, *species\_count*.  
Deprecated.
- Section 4.2: **eVTR Form Specification**
  - Added *vendor\_app\_platform* entry to specification.
  - Removed the following fields: *effort\_count*, *catch count*, *effort\_record\_count\_by\_trip*, *catch\_record\_count\_by\_subtrip*, *species\_count*.  
Deprecated.

## 7.6 Revision 1.4 Updates

Release Date: December 15, 2021

- Section 2.3.2
  - Updated year in UTID example.
- Section 2.3.3
  - Changed UTID text from “Must be available early in the trip, before the Trip Start Hail” to “Must be available when the trip report is initiated”.
- **Global edit:** Removed references to Catch and Sales data as one array.  
**Context:** Submitting an eVTR form for a logged in, authenticated user to the GARFO eVTR database.  
**Affected sections in the guide:**
  - Section 3.3 forward
- Section 3.3.3.2
  - Updated date field values to include “21” as year
  - Changed case in value: “*vendor\_app\_platform*”: “ANDROID”,
  - Deleted “*vtr\_serial\_num*:12345619063023,”
  - Deleted “*datetime\_haul\_start*: 1409190739,”
  - Deleted “*datetime\_haul\_end*: 1409190739,”
- Section 4.1
  - Updated description of support tables.
- Section 4.2
  - Deleted *vtr\_serial\_num*
  - Changed INTEGER to NUMERIC STRING in *chart\_area*
  - Changed INTEGER to NUMERIC STRING in *offloading\_port*
- Section 4.3
  - Updated description of *number\_of\_crew* field in [Table 2: Reporting Fields Based on Trip Activity Type](#).
- Section 4.5
  - Deleted RULE ID 32 (TRIP)
  - Added RULE ID 232 and 233 (CATCH)

## 7.7 *Revision 1.4.1 Updates*

Release Date: December 28, 2021

- Section 4.7.6
  - Added Release Date reference (does not affect technical contents of guide).

## 7.8 *Revision 1.5 Updates*

Release Date: April 15, 2022

- Section 2.3.2
  - Updated year in UTID example.
- Section 4.5
  - Added RULE ID: N/A (1)
  - Added RULE ID: 4 (TRIP)
  - Added RULE ID: 39 (TRIP)
  - Deleted RULE ID: 169 (TRIP)
  - Added RULE ID: 35 (SUBTRIP)
  - Added RULE ID: 36 (SUBTRIP)
  - Added RULE ID: 24 (CATCH)
  - Added RULE ID: 234 (CATCH)

## 7.9 *Revision 2.0 Updates*

Release Date: March 29, 2024

- Section 2.3.2
  - Updated year UTID example.
- Section 4.5
  - Updated Quality Control Audit Rules
- Section 4.6 (New)
  - Added Quality Control Audit Rules for Lobster and Crab
- Section 5.0 (New)
  - Added instructions for organizations that offer eVTR applications that include American Lobster fishing trips
- Section 6.0 (New)
  - Added instructions for organizations that offer eVTR applications that include clam trips.
- Section 7.1.9.1
  - Added table showing the updates for the Quality Control Audit Rules from Rev. 1.5 of this guide to Rev. 2.0 of this guide.

[Jump to: Table of Contents](#)

## 7.9.1 Updates to Quality Control Audit Rules

[Table 7](#) shows how the Quality Control audit rules have been updated from Rev. 1.5 of this guide (April 2022) to Rev. 2.0 of this guide (March 2024).

**Table 7: Updates to Audit Rules from Rev. 1.5 to Rev. 2.0 of This Guide**

Level	Rule ID Listed in Rev. 1.5	Rule Name	Rev. 2.0 Action based on Updated Rules	Rule Description Change	Rule Detail Change
1	Not Applicable	VALID OPERATOR PERMIT NUMBER	Removed	Not Applicable	Not Applicable
TRIP	3	DATE LAND BEFORE DATE SAIL	NO CHANGE	Not Applicable	Not Applicable
TRIP	4	DATE LAND AFTER TODAYS DATE	Changed	YES	YES
TRIP	5	DATE SAIL AFTER TODAYS DATE	NO CHANGE	Not Applicable	Not Applicable
TRIP	14	TRIP ID DOES NOT MATCH VESSEL PERMIT NUMBER	NO CHANGE	Not Applicable	Not Applicable
TRIP	23	TRIP MISSING SUBTRIP RECORDS	Changed	YES	NO
TRIP	Not Applicable	TRIP MISSING CATCH DATA	Rule ID = 24 (NEW AUDIT RULE)	Not Applicable	Not Applicable
TRIP	27	VESSEL PERMIT NUMBER SUBMITTED INVALID	Removed	Not Applicable	Not Applicable
TRIP	34	DUPLICATE OF EXISTING TRIP WITH SAME PERMIT AND DATE SAIL	Removed	Not Applicable	Not Applicable
TRIP	Not Applicable	INVALID NUMBER OF ANGLERS	Rule ID = 38 (NEW AUDIT RULE)	Not Applicable	Not Applicable
TRIP	39	SOAK TIME EXCEEDS LENGTH OF TRIP	Removed	Not Applicable	Not Applicable
TRIP	Not Applicable	DUPLICATE OF EXISTING TRIP WITH SAME PERMIT AND DATE SAIL	Rule ID = 40 (NEW AUDIT RULE)	Not Applicable	Not Applicable
TRIP	69	TRIP DATE SAIL OVERLAPS AN EXISTING TRIP	Changed	Yes	No
TRIP	155	INVALID TEST EVTR SUBMISSION	Removed	Not Applicable	Not Applicable

Level	Rule ID Listed in Rev. 1.5	Rule Name	Rev. 2.0 Action based on Updated Rules	Rule Description Change	Rule Detail Change
TRIP	160	INVALID TRIP TYPE	Removed	Not Applicable	Not Applicable
TRIP	161	INVALID OPERATOR NAME LENGTH	NO CHANGE	Not Applicable	Not Applicable
TRIP	168	MINIMUM TRIP LENGTH NOT MET	NO CHANGE	Not Applicable	Not Applicable
TRIP	Not Applicable	INVALID NUMBER OF ANGLERS	Rule ID = 175 (NEW AUDIT RULE)	Not Applicable	Not Applicable
TRIP	177	INVALID OPERATOR NAME	Changed	NO	YES
TRIP	179	INVALID NUMBER OF CREW	Removed	Not Applicable	Not Applicable
TRIP	Not Applicable	TRIP CONTAINS SPLIT EFFORT	Rule ID = 182 (NEW AUDIT RULE)	Not Applicable	Not Applicable
TRIP	Not Applicable	INVALID NUMBER OF CREW	Rule ID = 215 (NEW AUDIT RULE)	Not Applicable	Not Applicable
TRIP	Not Applicable	NOT A PRODUCTION USER	Rule ID = 231 (NEW AUDIT RULE)	Not Applicable	Not Applicable
TRIP	Not Applicable	CATCH QUANTITY NOT EQUAL TO SOLD QUANTITY	Rule ID = 232 (NEW AUDIT RULE)	Not Applicable	Not Applicable
TRIP	Not Applicable	SOLD QUANTITY NOT EQUAL TO CATCH QUANTITY	Rule ID = 233 (NEW AUDIT RULE)	Not Applicable	Not Applicable
SUBTRIP	6	GEARCODE INVALID	Changed	NO	YES
SUBTRIP	7	AREA INVALID	Changed	NO	YES
SUBTRIP	8	GEAR REPORTED WITHOUT A SPECIES	NO CHANGE	Not Applicable	Not Applicable
SUBTRIP	21	SUBTRIP/HAUL MISSING CATCH RECORDS	Removed	Not Applicable	Not Applicable
SUBTRIP	22	ORPHAN SUBTRIP RECORD	NO CHANGE	Not Applicable	Not Applicable
SUBTRIP	Not Applicable	SET ONLY TRIP MISSING REQUIRED VALUES	Rule ID = 30 (NEW AUDIT RULE)	Not Applicable	Not Applicable



Level	Rule ID Listed in Rev. 1.5	Rule Name	Rev. 2.0 Action based on Updated Rules	Rule Description Change	Rule Detail Change
SUBTRIP	33	MISMATCH BETWEEN CATCH REPORTED AND CATCH RECORD COUNT	Removed	Not Applicable	Not Applicable
SUBTRIP	35	INVALID SOAK TIME FOR GEAR CODE	Changed	NO	YES
SUBTRIP	36	INVALID GEAR QUANTITY OR NUMBER OF HAULS	Changed	NO	YES
SUBTRIP	Not Applicable	SOAK TIME EXCEEDS LENGTH OF TRIP	Rule ID = 39 (NEW AUDIT RULE)	Not Applicable	Not Applicable
SUBTRIP	162	INVALID AREA LATITUDE LONGITUDE	NO CHANGE	Not Applicable	Not Applicable
SUBTRIP	166	NUMBER OF HAULS MISSING	NO CHANGE	Not Applicable	Not Applicable
SUBTRIP	171	INVALID GEAR VALUES FOR PTL	REMOVED	Not Applicable	Not Applicable
SUBTRIP	Not Applicable	INVALID FISHING DEPTH	Rule ID = 176 (NEW AUDIT RULE)	Not Applicable	Not Applicable
SUBTRIP	180	MESH SIZE MISSING OR INVALID VALUE FOR GEAR TYPE	Changed	NO	YES
SUBTRIP	181	GEAR SIZE MISSING OR INVALID VALUE FOR GEAR TYPE	Changed	NO	YES
CATCH	9	SPECIES CODE INVALID	Changed	NO	YES
CATCH	15	INVALID PORT CODE	Removed	Not Applicable	Not Applicable
CATCH	16	PORT CODE IS MISSING	Removed	Not Applicable	Not Applicable
CATCH	20	ORPHAN CATCH RECORD	Removed	Not Applicable	Not Applicable
CATCH	24	TRIP MISSING CATCH DATA DETAIL	Removed	Not Applicable	Not Applicable
CATCH	25	INVALID DEALER NUMBER	Removed	Not Applicable	Not Applicable
CATCH	28	SPECIES CATCH WEIGHT MISSING IN KEPT AND DISCARDED	Changed	NO	YES
CATCH	72	NO CATCH REPORTED A DEALER	NO CHANGE	Not Applicable	Not Applicable

Level	Rule ID Listed in Rev. 1.5	Rule Name	Rev. 2.0 Action based on Updated Rules	Rule Description Change	Rule Detail Change
CATCH	232	SOLD QUANTITY NOT EQUAL TO CATCH QUANTITY	Removed	Not Applicable	Not Applicable
CATCH	233	CATCH QUANTITY NOT EQUAL TO SOLD QUANTITY	Removed	Not Applicable	Not Applicable
CATCH	234	DATE SOLD BEFORE DATE LAND	Removed	Not Applicable	Not Applicable
SALE	Not Applicable	SPECIES CODE INVALID	Rule ID = 9 (NEW AUDIT RULE)	Not Applicable	Not Applicable
SALE	Not Applicable	INVALID DEALER NUMBER	Rule ID = 183 (NEW AUDIT RULE)	Not Applicable	Not Applicable
SALE	Not Applicable	DATE SOLD MISSING	Rule ID = 184 (NEW AUDIT RULE)	Not Applicable	Not Applicable
SALE	Not Applicable	INVALID PORT CODE	Rule ID = 229 (NEW AUDIT RULE)	Not Applicable	Not Applicable
SALE	Not Applicable	PORT CODE IS MISSING	Rule ID = 230 (NEW AUDIT RULE)	Not Applicable	Not Applicable
SALE	Not Applicable	DATE SOLD BEFORE DATE LAND	Rule ID = 234 (NEW AUDIT RULE)	Not Applicable	Not Applicable

[Jump to: Table of Contents](#)