

OpenTrack Add API Method

OpenTrack is a Data Exchange application that allows outside vendors to access your information for data services. In order for a vendor to access your data, an authorized user will need to define the vendor's settings for each API (Application Programming Interface) they intend to access. (There are six standard APIs that can be used for connecting to the DMS to retrieve data). This document will walk you through the setups for allowing a vendor to access your data via OpenTrack.

You can add a new API Method by taking function **Add** from the [OpenTrack](#) option on the [Data Exchange](#) application menu.

Contents

[\[hide\]](#)

- [1 User Guide](#)
- [2 QuickStart Guides](#)
- [3 Use Cases](#)

User Guide

To authorize a vendor to your data using OpenTrack , take the following steps:

1. From the [Data Exchange](#) application menu, take menu option [OpenTrack](#). The Active Methods list screen will display.
 1. If the OpenTrack menu option does not appear on the Data Exchange application menu, the user security will need to be adjusted.
 1. The “Authorize to OpenTrack” prompt can be found in the [Data Extraction](#) section of the [User Security](#).
2. Use function **Add** on the Active Methods list screen. The Vendor Access definition screen will display.



Vendor Access definition screen

3. Define the vendor specifications:
 1. **Method** – Select the desired API method from the Available Methods list screen.



Available Methods list screen

1. Available Methods are defined by the Dealertrack DMS. Available methods include:

1. Customers
2. Vehicles
3. Deals
4. Accounting
5. Service
6. Parts

2. **Vendor** – Select the desired vendor from the Certified Vendors list screen.



Certified Vendors list screen

1. Vendors must be OpenTrack certified through the Dealertrack DMS before they appear on this list

2. Only vendors that correspond to the selected Method will display.

3. **Access Begins** – Key in the date for vendor access to begin in the mmddyy format, without dashes or slashes.



Access Begins/Access Ends

1. When a request for data access is received via our web service, it will be validated against the Access Begins date. Vendor Requests before this date will not be honored.

4. **Access Ends** – Key in the date for vendor access to end in the mmddyy format, without dashes or slashes.

1. When a request for data access is received via our web service, it will be validated against the Access Ends date. Vendor Requests after this date will not be honored.

2. Only use this field if you wish the vendor's access to be stopped at a future date.

3. This field can be left blank.

4. Use function **OK**. Your vendor will be added to the list of API Methods.



Options and Functions

1. A check will be performed to ensure that the combination of vendor and method does not already exist in the Active Methods list as an active method.
 1. If the vendor/method combination already exists, the message will display:
"Access to this method for the selected vendor already exists."
2. If your **Access Begins** date is set later than the current date or if the **Access Ends** date has already passed, you will need to use function **Inactive** to toggle between your Active and Inactive methods.
 1. Methods are considered inactive when they have not yet reached the active date or when they have passed end date.
 2. Multiple instances of the same inactive vendor/method combination can exist. The purpose for this is to retain history.
3. If you need to change the vendor specifications, use option **Deactivate** on the vendor line.
 1. You can change the **Access Begins** or **Access Ends** date.
 2. You cannot change the **Method** or the **Vendor**. Changing the Method or Vendor requires setting up a new API Method.
 3. If you wish to remove an API Method, simply key in the **Access Ends** date for the appropriate date. The method will be moved from your Active Methods list to your Inactive Methods list. **If at a later date you wish to reactive the method, you can just key in a new Access Begins date.**
5. Repeat this process for each API method.