USER GUIDE

VisionLink® VL-Ready API

VLR_API_01_0420
April 2020

VISIONLINK.

Corporate Office

VirtualSite Solutions LLC
10368 Westmoor Drive
Westminster, CO 80021
USA

Copyright and Trademarks

Release Notice

This is the April 2020 release (VLR_API_01_0420) of the **VisionLink® VL-Ready API User Guide**. This **VisionLink VL-Ready API User Guide** replaces the **VisionLink VL-Ready API User Guide** published May 2016 and earlier.

Contents

Introduction

This user guide provides an overview of the VisionLink® software VL-Ready API. After reading this document, you will:

- Understand VL-Ready API, and what each API provides
- Know where to get help with VL-Ready API
- Know howto access the VL-Ready API

About VL-Ready API

VL-Ready API provide a means to use your VisionLink software-related data in a data-feed format designed for easy access by third-party tools. VL-Ready API are typically used by dealers, customers, and integrators who want to integrate VisionLink software-related data in their ERP, accounting, or other third-party systems. These API provide:

- A source for a rich set of telematics and operational data that are published in an easy-to-consume format.
- Access to Fleet Summary and Asset Operation information for manipulating groups, users, and alert configurations.
- Data in a uniform pattern that can be easily integrated within other applications, and can be tested.

Note – *VL-Ready API provide the most recent 14 days of data. This two-week period is a rolling window.*

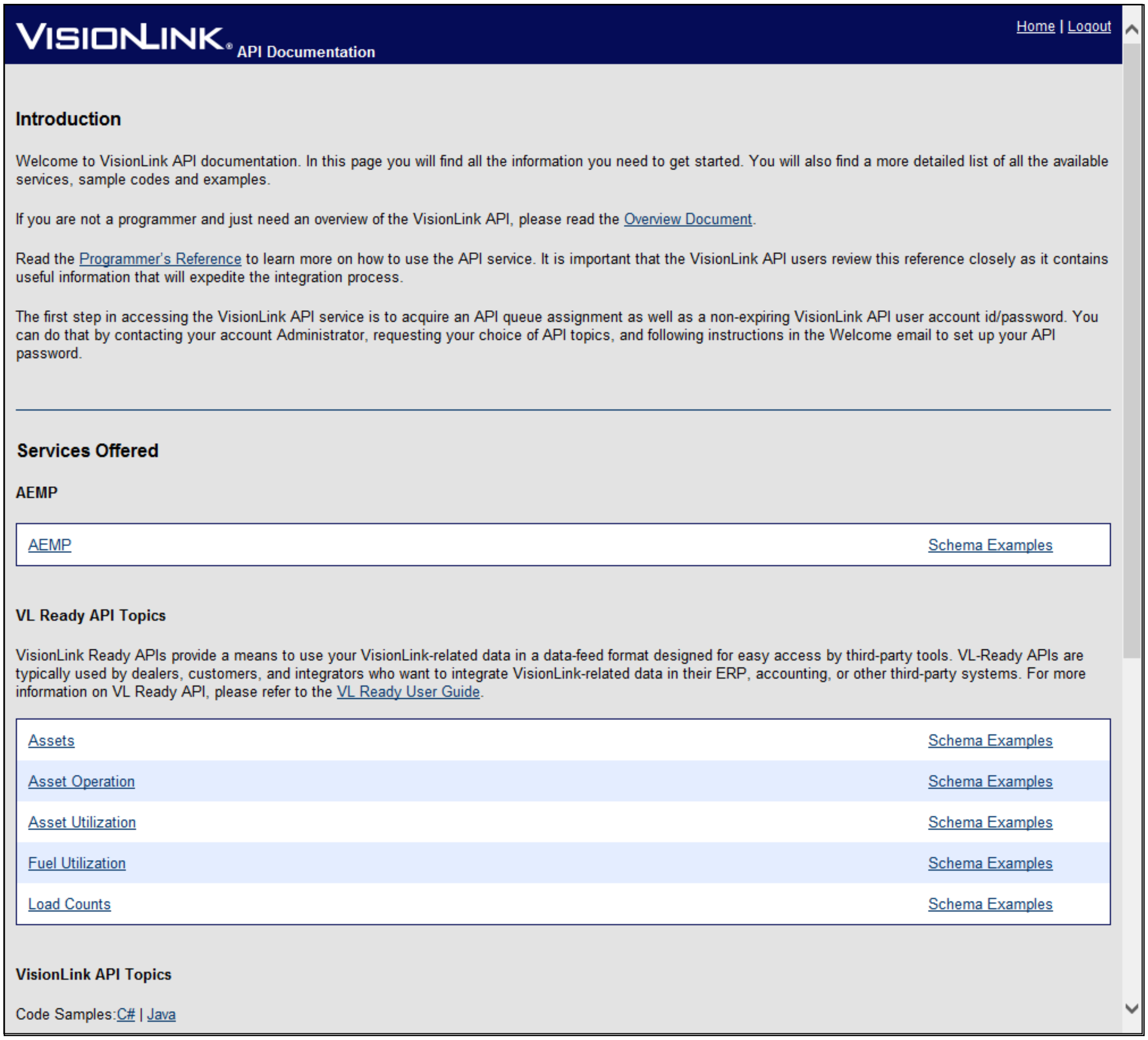
Requesting access to VL-Ready API

Follow this procedure to gain access to the VisionLink API services for the first time:

1. Acquire an API queue assignment and request a non-expiring VisionLink API user account id/password from your VisionLink account administrator.
2. Request your choice of API topics from the VisionLink account administrator.
3. Follow the instructions in the Welcome email to create your API password.

An automatic email notifies you once you are granted access to the API. In the future, request access to additional API topics by contacting your VisionLink account administrator.

To access and manage queues, go to https://legacy-apidocumentation.myvisionlink.com:



Introduction

Welcome to VisionLink API documentation. In this page you will find all the information you need to get started. You will also find a more detailed list of all the available services, sample codes and examples.

If you are not a programmer and just need an overview of the VisionLink API, please read the Overview Document.

Read the Programmer's Reference to learn more on how to use the API service. It is important that the VisionLink API users review this reference closely as it contains useful information that will expedite the integration process.

The first step in accessing the VisionLink API service is to acquire an API queue assignment as well as a non-expiring VisionLink API user account id/password. You can do that by contacting your account Administrator, requesting your choice of API topics, and following instructions in the Welcome email to set up your API password.

Services Offered

AEMP

AEMP	Schema Examples

VL Ready API Topics

VisionLink Ready APIs provide a means to use your VisionLink-related data in a data-feed format designed for easy access by third-party tools. VL-Ready APIs are typically used by dealers, customers, and integrators who want to integrate VisionLink-related data in their ERP, accounting, or other third-party systems. For more information on VL Ready API, please refer to the VL Ready User Guide.

Assets	Schema Examples
Asset Operation	Schema Examples
Asset Utilization	Schema Examples
Fuel Utilization	Schema Examples
Load Counts	Schema Examples

VisionLink API Topics

Code Samples:C# | Java

This site provides several resources:

- Documentation detailing the content of each APItopic.
- Links to XML schema (XSD) that provide access to the schemas describing the XML content for each topic. The schemas can be used with a validating XML parser to validate the content of the API XML.

VL-Ready API support

If you have a problem and cannot find the information you need in the product documentation, contact your local dealer (Cat®, SITECH®, or Trimble®) or your VisionLink account administrator.

Understanding the VL-Ready API capabilities

URLs and naming conventions

The VL-Ready API defines data endpoints via URL structures as resource identifiers. This is done to encourage discovery of new data feed-endpoints and ensure proper data-feed performance across the VL-Ready API solution.

VL-Ready API versions are immutable once they are published; however, you should always rely on the URLs reported by the data feed responses instead of hard-coding them because they are guaranteed to point to the proper URL for a specific resource.

Most endpoints provide links to far more data than you may need for your service, so you should access only the data that you require.

Security/Authentication and access

The VL-Ready API follow REST (REpresentational State Transfer) conventions and handle authentication through the use of HTTP Basic Authentication, which requires secure API login credentials and appropriate service plans and features enabled for the devices.

You access an API by requesting a URL using the API login credentials you were provided when you requested access.

Response formats

All VL-Ready requests provide XML-formatted responses, served as application/XML content. By default, content is returned with a charset of UTF-8 (ASCII). If any of your information (for example, Asset name) contains Unicode/double-byte characters, specify `accept-charset: utf-16` as a header in your VL-Ready request so that the content will be returned with a charset of UTF-16.

Response codes and affiliated errors

Response codes are included in the HTTP status response header. If an error occurs, in most cases the VisionLink software will provide a specific error status code in its HTTP status response to indicate details about the error. For example:

Status Code	Description
200	Success (upon a successful GET, PUT, or DELETE request)
400	Bad request (requesting something that is perceived to be a client error)
401	Unauthorized (incorrect or missing authentication credentials)
404	Resource Not Found (requesting a non-existent piece of data)
500	Application Error

While these messages are considered the most descriptive details about the error condition, they are subject to change with new versions of REST endpoints.

Common data in feeds

General structure

- Each feed starts with a list of URLs that indicates the current page of data, the page before, and the page after in a <Nav>tag.
- Asset details follow page information in an <Asset> tag. The information you find within that tag include asset-specific details including Make, Serial Number, etc. A URL is listed before each <Asset> tag. If you want more asset-specific information than is listed in the feed, use the URL provided (see the XML examples below).
- Feed-specific data is provided at the asset level. For example, the Asset Utilization API provides Utilization information for each asset within that asset's detail section. The Asset Utilization information provided in the Asset Utilization feed is identical to that shown in (and used by) Asset Utilization in the VisionLink software. If no Utilization information is available for a particular asset, no Asset Utilization section will be present in the feed.

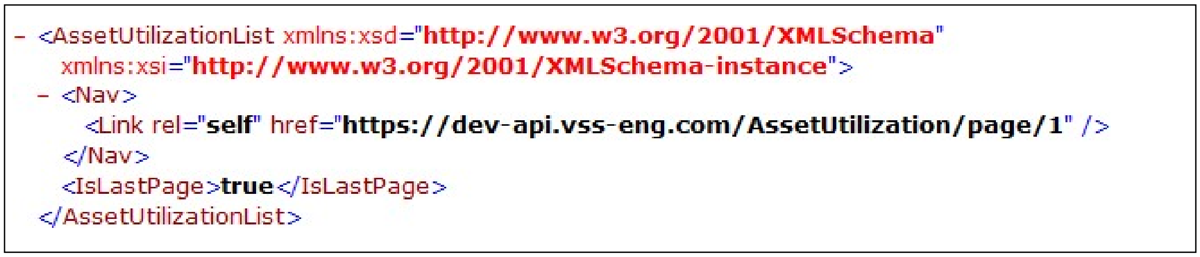
Pagination

The VL-Ready API uses pagination when sending its data. When a feed's endpoints return collections (more than one record) of data, the XML content will provide page number information as bookmarks.

The response contains a <IsLastPage> field that indicates when more records are available. It's located immediately below the </Nav> tag.

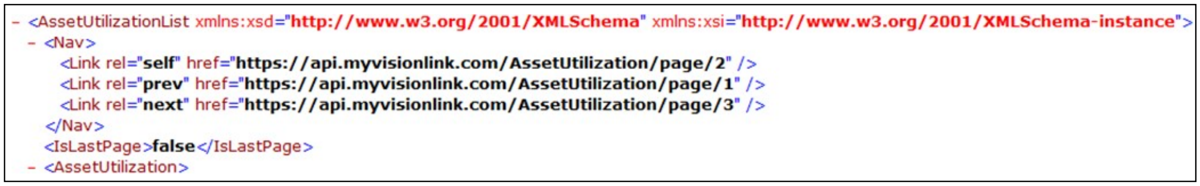
You will know that you have obtained all available data available in the XML record:

- When <IsLastPage>is "true," you have received all currently available data and that no additional data is available beyond what the current page contains:



```
- <AssetUtilizationList xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  - <Nav>
      <Link rel="self" href="https://dev-api.vss-eng.com/AssetUtilization/page/1" />
    </Nav>
    <IsLastPage>true</IsLastPage>
  </AssetUtilizationList>
```

- When <IsLastPage> is “false,” a link to the next page will be posted and identified as “next.” Enter the URL in your browser to request the next page of data (see below):



```
- <AssetUtilizationList xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  - <Nav>
      <Link rel="self" href="https://api.myvisionlink.com/AssetUtilization/page/2" />
      <Link rel="prev" href="https://api.myvisionlink.com/AssetUtilization/page/1" />
      <Link rel="next" href="https://api.myvisionlink.com/AssetUtilization/page/3" />
    </Nav>
    <IsLastPage>false</IsLastPage>
  - <AssetUtilization>
```

Note – *To ensure you get all the data, use the page numbers in each link to progress through the records provided by the feed until you encounter <IsLastPage>=”true”.*

Bookmarks

Bookmarks are strings that identify the next set of available data. As described above, VL-Ready uses pagination. When each page of data is loaded, you will be able to see which page you are on and which pages precede and follow that page.

Filtering

Filtering is a mechanism that allows retrieval of targeted information from VL-Ready feeds.

- For the Asset Operation feed, you can provide a timestamp (UTC; in ISO8601 format) as a filter parameter. By doing this, you will be able to obtain specific data that has been received by the VisionLink software **on or after** the date provided in the request URL.
- For the Asset Utilization and Fuel Utilization feeds, you can provide a date (in yyyy-MM-dd format, where yyyy=year, MM=month, dd=day) as a filter parameter. By doing this, you will be able to obtain specific data that has been received by the VisionLink software **on or after** the date provided in the request URL.

Note – *The service uses UTC to determine the 'Current' date and time. If the timestamp or date provided is older than 14 (24-hour periods) in the past, or if the timestamp or date provided is in the future, the URL will return HTTP 400 (Bad Request).*

Filtering on the 'Current' date may not result in a complete, 24-hour picture of your data. To get 24 hours of data, include a filter parameter for the previous day. It is not necessary to include a filter parameter for today's date, because the service has determined the 'Current' date and time. For example, if today is 30 June 2015:

https://[VisionLink Environment Base URL]/APIService/VLReady/AssetUtilization?FromAssetLocalDate={2015-06-29}

Units of measure

Some data fields contain a unit of measure that will help you understand, interpret, and use the data. In these cases, the unit of measure is identified by the <uom> tag.

Messaging

In the Utilization XSDs, you will see message URLs. These URLs can have supported and received attributes. An example is shown below:

```
<xs:complexType mixed="true">
<xs:sequence>
<xs:element type="xs:float" name="Value" minOccurs="0"/>
</xs:sequence>
<xs:attribute type="xs:string" name="supported" use="optional"/>
<xs:attribute type="xs:string" name="received" use="optional"/>
<xs:attribute type="xs:string" name="messageUrl" use="optional"/>
</xs:complexType>
```

A ‘supported’ attribute on an element indicates whether the current device installed on the asset supports sending the information. For example, <IdleHours supported=’false’> means that the device currently installed on the asset does not support/send IdleHours to the VisionLink software.

A ‘received’ attribute on an element indicates whether the VisionLink software has received the value for the particular type of information. For example, <IdleHours received=’false’> means that the VisionLink software has not received any IdleHours data for the given day. In addition, <IdleHours supported=’true’ received=’false’> means that the device supports IdleHours but the VisionLink software has not received any data for the given calendar day. Currently the UI indicates this information by ‘– ‘ and ‘N/A’ where ‘–‘ means that the value is not reported and ‘N/A’ means that the device does not support sending this piece of information.

The Asset Operation, Asset Utilization, and Fuel Utilization feeds currently provide the following messages:

Missing meter value detected

What this means:

The asset did not report a specific value for the given day. This situation can arise if:

- The device was outside of satellite or cellular coverage
- The device was unable to obtain specific information from the Electronic Control Module (ECM) at the time ofreporting into the VisionLink software

Example:

Asset ABC123 is supposed to report RuntimeHours every 6 hours. The VisionLink software receives a RuntimeHours report from this asset on Feb. 2, 2015, at 09:00:00 UTC. The next RuntimeHours report from this asset is received on Feb. 4, 2015, at 17:00:00 UTC.

The VisionLink software was supposed to receive a RuntimeHours report on Feb. 3. But that did not happen. In this case, when the Asset Utilization feed is executed on Feb. 3, 2015, the ‘RuntimeHours’ field will have a callout value = 1.

It will look similar to following snippet:

```
<RuntimeHours received="true" supported="true">
    <Callout
    messageUrl="https://api.myvisionlink.com/UtilizationMessage/1">
```

```
      <Value>1</Value>
   </Callout>
</RuntimeHours >
```

Impact on calculations:

Since the ‘RuntimeHours’ value is required to calculate ‘RunningUtilizationPercentage’ and ‘EfficiencyPercentage’, those fields will also have the same callout message. If these values from the Asset Utilization feed are being used in a specific manner, all integrating systems are advised to take this ‘missing data’ situation and adjust any downstream calculations.

Value includes data occurring outside of the specified day

What this means:

The asset fails to report a specific value for more than 1 day. This typically happens if the asset is out of cellular coverage or is at a place where it cannot make a satellite connection.

Example:

Asset ABC123 is supposed to report RuntimeHours every 6 hours. The VisionLink software receives a RuntimeHours report from this asset on Feb. 2, 2015. The next RuntimeHours report from this asset is received on Feb. 4, 2015, at 17:00:00 UTC.

The VisionLink software was supposed to receive a RuntimeHours report on Feb. 3. But that did not happen. In this case, when the Asset Utilization feed is executed on Feb. 4, 2015, at 22:00:00 UTC, the ‘RuntimeHours’ field will have a callout value = 2. This indicates that the ‘calculated delta’ value spans multiple days.

It will look similar to following snippet:

```
<RuntimeHours received="true" supported="true">
   <Callout
   messageUrl="https://api.myvisionlink.com/UtilizationMessage/2">
      <Value>2</Value>
   </Callout>
  <Value>22</Value>
</RuntimeHours >
```

Impact on calculations:

Since ‘RuntimeHours’ value is required to calculate ‘RunningUtilizationPercentage’ and ‘EfficiencyPercentage’, those fields will also have the same callout message. If these values from the Asset Utilization feed are being used in a specific manner, all integrating systems are advised to take this ‘missing data’ situation and adjust any downstream calculations.

Invalid value detected

What this means:

The asset has reported a value that is obviously erroneous. This typically happens due to a glitch in the way asset is sampling sensor data. In most cases, the asset follows up by reporting a valid value.

Example:

Asset ABC123 reports a ‘RuntimeHours’ of “25” on Feb. 4. It then reported a ‘RuntimeHours’ value of “50” on Feb. 5. Since an asset cannot run for 25 hours in a 24 hour period, this value is wrong.

It will look similar to the following snippet:

```
<RuntimeHours received="false" supported="true">
   <Callout
   messageUrl="https://api.myvisionlink.com/UtilizationMessage/3">
      <Value>3</Value>
   </Callout>
</RuntimeHours>
```

Impact on calculations:

Since 'RuntimeHours' value is required to calculate 'RunningUtilizationPercentage', that field will also have the same callout message. If these values from the Asset Utilization feed are being used in a specific manner, all integrating systems are advised to take this 'missing data' situation and adjust any downstream calculations.

Insufficient Runtime meter precision for valid Calculation

What this means:

The asset has reported RunTimeHours and IdleHours with a resolution which will generate negative 'WorkingHours' value. This callout only applies to the 'WorkingHours' field (which otherwise always contains the same callout type as IdleHours). It will look similar to the following snippet:

```
<WorkingHours received="false" supported="true">
   <Callout
   messageUrl="https://api.myvisionlink.com/UtilizationMessage/5">
      <Value>5</Value>
   </Callout>
</WorkingHours>
```

These messages are significant because the VisionLink software calculates its utilization based on information reported by the device. There are instances, however, when the device (for whatever reason) does not report values for a particular day or reports values with precision that is insufficient to accurately calculate utilization. These messages exist to inform the consumer about such a situation. Currently our UI shows this information by coloring the grid borders red. When you hover the cursor over the cell/value in the UI, these messages are shown. This information helps keep the user apprised of the state of device data so that informed decisions can be made.

Rate limiting

Unless otherwise specified by the endpoint documentation, the VisionLink software allows you to make one (1) request per second. Just because you are allowed to make one request per second doesn't mean you should. In fact, if your application is making API requests too quickly, the URL will return HTTP 401 Status Code.

***Note –** As a best practice, assess the frequency of access needed for your software implementation, and request the data only as you need it.*

Current VL-Ready API

Each VL-Ready API specification contains a list of URLs with their supported HTTP methods and a description of their behavior. Some VL-Ready API endpoints have a Request and Response format and a field listing that details the feed contents. The VisionLink software currently offers the following VL-Ready API:

Assets:	https://api.myvisionlink.com/Assets
Asset Operation:	https://api.myvisionlink.com/AssetOperation
Asset Utilization:	https://api.myvisionlink.com/AssetUtilization
Fuel Utilization:	https://api.myvisionlink.com/FuelUtilization
Load Counts:	https://api.myvisionlink.com/LoadCount

Each data feed is described on the following pages. Access the sample schemas from the API Documentation web page.

Assets

The Assets endpoint provides asset information for the requested customer.

URL structure

https://[VisionLink Environment Base URL] /APIService/VLReady/Assets/{pageNumber}
Gets the set of asset information records for your fleet.
HTTP Method: GET

Field listing

Asset Summary (<Assets>)	
URL attribute	A link that takes you to more details about this asset.
VisionLink Identifier	A unique identifier for the asset.
Make Code	The make code for this asset.
Make Name	The specific name of the asset.
Serial Number	The equipment serial number.
Asset ID	The asset name.

Model	The equipment model.
Equipment VIN	The vehicle identification number (VIN) of this asset (if applicable).
Product Family	The family of the product an asset belongs to (Example: Track type Tractor).
Manufactured Year	Year the asset was manufactured in.
Device Type	Type of the device installed on the asset.
Device Serial Number	The serial number of the device installed on the asset.

Asset Operation

The Asset Operation endpoint provides asset and asset operation information similar to that shown in Utilization->Asset Operation in the VisionLink software. The asset operation information is calculated based on the specific work definition identified by the user.

The Asset Operation API provides data for the past 14 days. This 14-day period is a rolling window. Or if you prefer, you can provide a timestamp (UTC; in ISO8601 format) as a filter parameter.

To obtain data through this feed, assets must have active Basic and Utilization service plans for the given period.

***Note –** In the Asset Operation feed, the Start/Stop times are paired. Therefore, the idling and working durations will only appear after the asset has stopped that particular operation. The 'StartStateAssetLocalTime' and 'EndStateAssetLocalTime' field provided in the feed are calculated from the position (Latitude/Longitude) information reported by the asset along with Start/Stop information. If position information cannot be paired with a Start/Stop event, the VisionLink software will attempt to use the 'Last Reported Position' to calculate the 'AssetLocalTime'. If the asset has never reported a position, the VisionLink software will assume 'UTC' to be the default local time zone for the asset.*

URL structure (without Filtering)

https://[VisionLink Environment Base URL] /APIService/VLReady/AssetOperation/page/{pageNumber}
Gets all Asset Operation records for your fleet for the past 14 days. No filtering is applied.
HTTP Method: GET

URL structure (with Filtering)

https://[VisionLink Environment Base URL] /APIService/VLReady/AssetOperation?LastBookmarkUTC={ISO8601 timestamp in

UTC}

https://[VisionLink Environment Base URL] /APIService/VLReady/AssetOperation/page/{pageNumber}?LastBookmarkUTC={ISO8601 timestamp in UTC}

Gets the set of Asset Operation records for your fleet that has been received by the VisionLink software on or after the LastBookmarkUTC provided in the request.

HTTP Method: GET

Field listing

These tables represent the field-level detail contained in the Asset Operation endpoint.

Asset Operation (<AssetOperation>)	
Asset Summary	A summary of the asset information, and a link to more details.
Calendar Day Asset Local Time	The calendar date the asset operation is reported in asset local time zone.
Start State UTC	Operation start date and hour in UTC.
Start State Asset Local Time	Operation start date and hour in the asset local time zone.
Start Location	Location (lat/lon) of the asset when the start event was reported
End State UTC	Operation end date and hour in UTC.
End State Asset Local Time	Operation end date and hour in the asset local time zone.
End Location	Location (lat/lon) of the asset when the stop event was reported.
Duration	Duration of the operation in hours.
Work Definition	Determines how the Utilization information is calculated. Possible values include: • Movement Events • Sensor Events • Movement and Sensor EventsMeter Delta
Working State	Specifies the state the asset was in when Asset Operation information was calculated. Possible values are: • No Device State

- Awaiting First Report
- Asset On
- Idling
- Asset Off
- Reporting
- Not Reporting
- On Site
- Working
- Stopped
- Running
- Cool Down
- Engine Stopping
- Not Ready to Run
- Not Available

Asset Utilization

The Asset Utilization API returns asset utilization information for the customer. The asset utilization information is calculated based on the specific work definition identified by the user. The API provides data for the past 14 days, up to 100 assets per page.

- If no data exists for a particular asset, the AssetUtilization node for that asset will not be present.
- If no data exists for a particular day, the AssetUtilizationDetail node for that day will not be present.

The Asset element contains fiveelements:

- VisionLinkIdentifier
- MakeCode
- SerialNumber
- AssetID
- Model

The element also contains a 'URL' attribute that contains a link to detailed asset information.

To obtain data through this feed, assets must have active Basic and Utilization service plans for the given period.

URL structure (without Filtering)

https://[VisionLink Environment Base URL]/APIService/VLReady/AssetUtilization/
{pageNumber}

Gets the set of Asset Utilization records for your fleet.

HTTP Method: GET

URL structure (with Filtering)

You can provide a date (in yyyy-MM-dd format, where yyyy=year, MM=month, dd=day) as a filter parameter. By doing this, you will be able to obtain specific data that has been received by the VisionLink software **on or after** the date provided in the request URL.

https://[VisionLink Environment Base URL] /APIService/VLReady/AssetUtilization?FromAssetLocalDate={yyyy-MM-dd}

https://[VisionLink Environment Base URL] /APIService/VLReady/AssetUtilization/page/{pageNumber}?FromAssetLocalDate={yyyy-MM-dd}

Gets the set of Asset Utilization records for your fleet on or after the FromAssetLocalDate provided in the request.

HTTP Method: GET

Field listing

These tables represent the field-level detail contained in the Asset Utilization endpoint.

Asset Utilization (<AssetUtilization>)	
Asset Summary	A summary of the asset information, and a link to more details.to more details.
Date	Date the Asset Utilization is reported. Only the calendar date is reported.
IdleHours	Hours the asset was Idling for the specific date mentioned in that message.
WorkingHours	Hours the asset was working for the specific date mentioned in that message.
RuntimeHours	Runtime hours of the asset (idling + working) for the specific date mentioned in that message.
Work Definition	Determines how the Utilization information is calculated. Possible values include: • Movement Events • Sensor Events • Movement and Sensor Events • Meter Delta (Data obtained from the vehicle's diagnostic network)

Callouts	Special notes about a particular Utilization calculation. Possible values include: • Missing Meter Value • Multiple Day Delta • Spike • Not Applicable • Negative Value • No Data
Working Utilization Percentage	Percentage of time the asset was working based on the work definition defined for the specific date mentioned in that message.
Running Utilization Percentage	Percentage of time the asset was running based on the work definition defined for the specific date mentioned in that message.
Expected runtime hours	The number of hours the asset is expected to run for that day for the specific date mentioned in that message.
Efficiency percentage	Efficiency of the asset for the specific date mentioned in that message.

Fuel Utilization

The Fuel Utilization feed returns fuel utilization information for the customer. The fuel utilization information is calculated based on the specific work definition identified by the user. The API provides data for the past 14 days, up to 100 assets per page.

- If no data exists for a particular asset, the FuelUtilization node for that asset will not be present.
- If no data exists for a particular day, the FuelUtilizationDetail node for that day will not be present.

The Asset element contains fiveelements

- VisionLinkIdentifier
- MakeCode
- SerialNumber
- AssetID
- Model

The element also contains a ‘URL’ attribute that contains a link to detailed asset information.

To obtain data through this feed, assets must have active Basic and Utilization service plans for the given period.

URL structure (without Filtering)

https://[VisionLink Environment Base URL]/APIService/VLReady/FuelUtilization/{pageNumber}
Gets the set of Fuel Utilization records for your fleet.
HTTP Method: GET

URL structure (with Filtering)

You can provide a date (in yyyy-MM-dd format, where yyyy=year, MM=month, dd=day) as a filter parameter. By doing this, you will be able to obtain specific data that has been received by the VisionLink software **on or after** the date provided in the request URL.

https://[VisionLink Environment Base URL]/APIService/VLReady/FuelUtilization?FromAssetLocalDate={yyyy-MM-dd}
https://[VisionLink Environment Base URL]/APIService/VLReady/FuelUtilization/page/{pageNumber}?FromAssetLocalDate={yyyy-MM-dd}
Gets the set of Fuel Utilization records for your fleet that has been received by the VisionLink software on or after the FromAssetLocalDate provided in the request.
HTTP Method: GET

Field listing

These tables represent the field-level detail contained in the Fuel Utilization endpoint.

Fuel Utilization (<FuelUtilization>)	
Asset Summary	A summary of the asset information, and a link to more details.
Date	Date the Asset Operation is reported. Only the calendar date is reported.
IdleFuelBurnedGallons	The amount of fuel burned (in gallons) while the asset was idling for the specific date mentioned in that message.
IdleHours	Hours the asset was Idling for the specific date mentioned in that message.
IdleFuelBurnRate	Rate at which the Fuel was burning while the asset was idling for the specific date mentioned in that message.
Work Definition	Determines how the Utilization information is

	calculated. Possible values include: • Movement Events • Sensor Events • Movement and Sensor Events • Meter Delta
WorkingFuelBurnedRateGallons	Rate at which the fuel was burning while the asset was working (in Gallons) for the specific date mentioned in that message.
WorkingHours	The hours the asset has worked for the specific date mentioned in that message.
Callouts	Special notes about a particular Utilization calculation. Possible values include: • Missing Meter Value • Multiple Day Delta • Spike • Not Applicable • Negative Value • No Data
RuntimeFuelBurnedGallons	Rate at which the fuel was burning while the asset was running (working+idling) for the specific date mentioned in that message.
RuntimeHours	Hours at which the asset was running (working + idling) for the specific date mentioned in that message.

Load Counts

The Load Counts feed delivers all relevant information about load counts based on 2D Project Monitoring. It includes lat/long, geofence boundaries, date/time, asset ID, and distance travelled for load and dump events. The API provides data for the past 14 days, up to 100 assets per page.

The Asset element contains five elements:

- VisionLinkIdentifier
- MakeCode
- SerialNumber

- AssetID
- Model

To obtain data through this feed, assets must have active Basic, Utilization, and 2D Project Monitoring service plans for the given period.

URL structure

https://[VisionLink Environment Base URL]/APIService/VLReady/LoadCount/{pageNumber}
Gets the set of load count records for your fleet.
HTTP Method: GET

Field listing

These tables represent the field-level detail contained in the Load Count endpoint.

Load Count (<LoadCount>)	
Asset Summary	A summary of the asset information, and a link to more details.
LoadUtc	The calendar date and time the load event begain in asset local time zone.
LoadSiteName	The site (geofence name) where the load event was reported.
LoadLatitude	Location (latitude) of the asset when the load event was reported.
LoadLongitude	Location (longitude) of the asset when the load eventwas reported.
DumpUtc	The calendar date and time the dump event begain in asset local time zone.
DumpSiteName	The site (geofence name) where the dump event was reported.
DumpLatitude	Location (latitude) of the asset when the dump eventwas reported.
DumpLongitude	Location (longitude) of the asset when the dump event was reported.
LoadQuality	Load/dump event (for example, Site to Site; Switch to Switch; Switch to Site; Site to Switch)

Note – *Load and dump events correspond to the values used to generate the Load Count report in the VisionLink software.*

Correlating data from multiple VL-Ready API

The data from each VL-Ready API can be correlated to data from other API. Common fields exist in each API that can be used to correlate the data:

- <Asset> All VL-Ready API use this tag for the data they report per asset. To correlate Fuel Utilization and Asset Utilization data for a specific asset, search by any of the asset-specific information supported in the Asset API (S/N, Make, Model, etc.) to capture the data for that asset.

- <Date> For specific dates, you can search by the <Date> field and capture all data that has been reported on a specific date. For example, you can look up a duration of an asset operation for a specific asset on a specific date (using the Asset Operation API) and compare it to the Utilization of that same asset for that time period (using the Asset Utilization API).