

# TomTom Traffic Flow Bulk Feed - I95 Interface Specification

1. Introduction
2. Exchange
  - 2.1. Payload formats
  - 2.2. Freeflow parameters
  - 2.3. Recommendations for HTTP requests
    - 2.3.1. Use of If-Modified-Since
    - 2.3.2. Use of gzip compression
3. DATEX II payload
  - 3.1. Schema extension
  - 3.2. Content description
    - 3.2.1. freeflow
    - 3.2.2. travelTime and averageSpeed
    - 3.2.3. freeFlowTravelTime and freeFlowSpeed
    - 3.2.4. supplierCalculatedDataQuality
4. ProtoBuf payload
  - 4.1. TrafficFlowGroup
    - 4.1.1. MetaInformation
    - 4.1.2. SupplierAndClientInfo
  - 4.2. TrafficFlow
    - 4.2.1. Speed
    - 4.2.2. SpeedWithTimeStamp
    - 4.2.3. SectionSpeed
  - 4.3. Location
  - 4.4. ProtoBuf data structure
5. Geographical representation
  - 5.1. TMC
    - 5.1.1. referenceSettings
    - 5.1.2. TMC link identifier
    - 5.1.3. TMC direction
    - 5.1.4. Internal and external road segments
    - 5.1.5. Examples
    - 5.1.6. Predefined Location Product
  - 5.2. OpenLR

Version Number	Date	Comments	
1.5	October 16th	2012	Added HDF ProtoBuf
1.6	October 8th	2013	Update of entry points and HDF ProtoBuf format
1.7	November 25th	2013	Improved description for direction used in TMC
1.8	April 10	2014	Added example for TMC with sections
1.9	June 16 - August 15 2014	2014	Add speed in MPH; improves description of average-speed / travel-time for road-closure
1.92	February 19	2015	Adjustments for I95

## 1. Introduction

TomTom Traffic flow bulk feed contains real-time travel times and speeds for segments, based on TomTom's Traffic technology, with several possible granularities. It is a bulk feed designed for server to server integration with traffic control center software, routing, navigation and mapping applications. This document describes the data exchange method and the payload description of the TomTom Traffic flow bulk feed interface. Containing large amounts of data, the way the transmission is conducted by client can improve the service quality since the flow feed is a pull service. The feed can be integrated efficiently using the ProtoBuf format. We also offer compatibility with the payload and exchange of the DATEX II standard.

## 2. Exchange

The data exchange methods of the feeds described in this document follow a REST architectural style. The service uses the protocol HTTP / HTTPS.

### 2.1. Payload formats

By default DATEX II (see <http://datext2.eu> for further details) is used as return format.

Alternatively, Protocol Buffers is a way of encoding structured data in an efficient yet extensible format. TomTom can provide data using proprietary ProtoBuf. See below section [ProtoBuf payload](#) and <http://code.google.com/p/protobuf> for more information on Protocol Buffers. ProtoBuf is highly recommended since it is more efficient for the transmission of data than XML based formats.

The data is collected by the customer (pull) from a dedicated URL using REST/XML over HTTP. The URL is build up for the different product versions with the following components:

```
http://<servername>/tsq/hdf/<name>/<apiKey>/content.xml<?parameter=value>
```

Or (only ProtoBuf):

```
http://<servername>/tsq/hdf/<name>/<apiKey>/content.proto<?parameter=value>
```

where <apiKey> is a client specific key. Every customer is given a dedicated API key to access the URL to collect the datafile. The API key provides access to one or more products for a specific time period. The API key forms part of the URL used in the file collection process. Security is arranged through this specific API key in combination with IP whitelisting. This means only prior whitelisted IP addresses or IP ranges can retrieve data using the specified API Key. Please contact your account manager or support to request IP whitelisting.

### 2.2. Freeflow parameters

Within this service, for every road segment an estimate is given whether traffic is flowing freely or whether it is congested. Bandwidth and the size of the payload can be reduced by requesting only non-freeflow information. This is the part of the payload that changes frequently, while the freeflow information is most of the time static. In this manner a customer can get the subset of the road segments that contain a non-freeflow speed or traveltime value. More about freeflow and non-freeflow is described in the payload section of this document. The parameter works as follows:

```
http://<servername>/tsq/hdf/<name>/<apiKey>/content.xml?flowType=nff  
http://<servername>/tsq/hdf/<name>/<apiKey>/content.proto?flowType=nff
```

will return the non-freeflow information only, i.e. the current speed and status of links that are currently not in freeflow state.

```
http://<servername>/tsq/hdf/<name>/<apiKey>/content.xml?flowType=ff  
http://<servername>/tsq/hdf/<name>/<apiKey>/content.proto?flowType=ff
```

will return the freeflow information only, i.e. the freeflow speed of all the links independently of the their current state.

If the <flowType> parameter is not used all information, including freeflow and non-freeflow segments are included in the response. No delta mechanism is supported.

When using this parameter, at the first request the freeflow information should be downloaded. Afterwards, non-freeflow can be downloaded each following request. The freeflow and non-freeflow data is related. The segments for both files should be in sync, otherwise there is a mismatch in the locations of the free flow data. Therefore, every time the non-freeflow information is downloaded it has to be checked that the non-freeflow data corresponds to the freeflow data that the user already has. This can be done by verifying the <clientID> within the <supplierAndClientInfo> message in ProtoBuf format; or the <nationalIdentifier> within <publicationCreator> in Datex2 format.

## 2.3. Recommendations for HTTP requests

Since flow feeds can be very big, it is recommended to optimize as much as possible the transmission of the information. In this manner the client actually obtains more up-to-date information.

For this reason, besides using ProtoBuf and non-freeflow information, the next additional recommendations are provided.

### 2.3.1. Use of If-Modified-Since

It is highly recommended the use of the standard HTTP header `If-Modified-Since` with the last value of `Last-Modified` received. Used properly it will avoid downloading several times the same content. See details HTTP/1.1 standard (RFC2616 Section 14.25)

### 2.3.2. Use of gzip compression

The TomTom Traffic Flow bulk feed supports gzip compression of all type of responses. However, responses will be compressed only when requester stated its support. This should be specified through the standard HTTP header `Accept-Encoding: gzip`. This is recommended to reduce the size of the payload.

## 3. DATEX II payload

DATEX II is a European standard for the exchange of Traffic and Travel Information. The TomTom Traffic flow bulk feed is based on DATEX II version 1.0. More information on DATEX II is available from the official website <http://datex2.eu> in the download section (archive 1.0). For the payload specification, data dictionary and user guide the following documents can be downloaded from the DATEX II website. You have to be subscribed as a user of the website.

- DATEX II User Guide version 1.0 (DATEXIIv1.0-UserGuide\_v1.0.pdf)
  - Standard DATEX2 XML schema version 1.0 (DATEXIISchema\_1\_0\_1\_0.zip)
  - A browseable DATEX II version 1.0 data dictionary (DATEXIIv1.0-DataDictionary\_v1.0.zip)
- The used Elaborated Data Publication is described in more detail in section 4.11 of the User Guide.

### 3.1. Schema extension

The XML schema for the TomTom Traffic flow bulk feed is `DATEXIISchema_1_0_1_0_HDT_HDF.xsd`. It is an extension of the standard DATEX II XML schema: `DATEXIISchema_1_0_1_0.xsd`, available from the DATEX II website.

The extension consists of the new `TravelTimeValueExtensionType` which provides some new elements such as `averageSpeed` or `roadClosure`, and the `ElaboratedDataExtensionType` which provides a description. If required, the extended XSD can be provided by your account manager or representative.

### 3.2. Content description

The travel times per segment are provided in an `ElaboratedDataPublication`. At any point in time an `ElaboratedDataPublication` contains a snapshot of the latest situation. No delta mechanism is supported. An elaborated data publication contains:

- `publicationTime`
- `publicationCreator`, meta data about the creator
- `headerInformation`
- `referenceSettings`
- a list of elaborated data items

Each elaborated data item has a unique id. In subsequent `elaboratedDataPublications` the elaborated data for the same segment will have the same unique id. Each elaborated data item contains a `TravelTimeValue` record, which extends the `BasicDataValue`.

The following information is provided in the `TravelTimeValue`:

Field Name	DATEX2 Description	Remark
<code>affectedLocation</code>	A definition of a stretch of road with a direction	The segment identifier, see <i>Geographical representation</i>
<code>travelTime</code>	Current average travel time in seconds based on real-time measurements between the defined locations in the specified direction	Only provided in case of congestion

averageSpeed	Current average speed on the affected location, in km/h, directly based on the travelTime and the length of the affectedLocation	Only provided in case of congestion; is an extension of TravelTimeValue
freeFlowTravelTime	The travel time in seconds which would be expected under ideal freeflow conditions	Only provided in case of free flow conditions
freeFlowSpeed	The free flow speed in km/h which would be expected under ideal freeflow conditions, corresponding to the freeFlowTravelTime	Only provided in case of free flow conditions
supplierCalculatedDataQuality	A measure of data quality assigned to the value by the supplier. 100% equates to perfect quality.	For explanation, see section 3.2.4
averageSpeedInMPH	Average speed in miles per hour	Extension of TravelTimeValue

### 3.2.1. freeflow

For every segment an estimate is made whether traffic is flowing freely or whether it is congested. If traffic is flowing freely, the actual travel time will depend on the type of vehicle and the speed the driver is choosing. The actual speeds on each segment can differ a lot. In that case the TravelTimeValue will contain the freeFlowSpeed and freeFlowTravelTime. If traffic is not flowing freely, it is considered congested, and a driver cannot choose the speed. In that case the TravelTimeValue will contain the travelTime and averageSpeed.

### 3.2.2. travelTime and averageSpeed

The travelTime contains an estimate of the real-time travel time between the start and the end of the affected location. It is the instantaneous travel time, which means that it is the sum of the travel time estimations on the individual road segments of the affected location. Note that this may differ from the travel time a driver is experiencing, travelling from the start to the end of the affected location, especially for longer segments. The estimation of the travelTime is based on both real-time measurements and historic data. It provides speeds by day of the week and the time of day. The averageSpeed is directly derived from the provided travelTime and the length of the affectedLocation. Therefore, the supplierCalculatedDataQuality also applies to this value. In case roadClosure flag is not used and the affected location is closed averageSpeed is set to 0 km/h and travelTime refers to the travel time under free flow conditions.

### 3.2.3. freeFlowTravelTime and freeFlowSpeed

The freeFlowTravelTime is directly derived from the freeFlowSpeed and the length of the affectedLocation. The freeFlowSpeed is the expected speed under free flow conditions, based on historic data. It is averaged over different vehicle types. The value is capped by the speed limit.

### 3.2.4. supplierCalculatedDataQuality

This field contains a quality measure for the provided travel time and speed. It is based on the following variables:

- the number of real-time measurements from devices driving the affected locations at the moment
- the standard deviation of the measurements
- the percentage of the affected location that is covered with measurements
- compatibility of measurements with historic data
- the age of the measurements

When a small number of real-time measurements are available historic data will be taken into account. The transfer function from real-time to historic travel times is smoothed and the compatibility variable takes into account whether this is a typical or atypical day, the spread between the historic and real-time measurements as well as a probability measure of a jam appearance for the affected location at measurement time. If the value of supplierCalculatedDataQuality is below 60% then the historic data is taken into account. A value of 50% implies that the provided travel time is based only on the historic speed profile.

## 4. ProtoBuf payload

The payload can also be formatted using Protocol Buffers, in particular the TomTom's **ProtoBuf** format. TomTom will extend and maintain backwards compatibility of this format.

Protocol Buffers is a de-facto standard to encode and transmit any structured information in a compact and platform independent way. After defining the data structure an encoder and a decoder can be generated for integration into C++, Java and several other languages. See also: <http://code.google.com/p/protobuf> or <https://developers.google.com/protocol-buffers>.

The data structure of the Protocol Buffer output is defined by a .proto file. **ProtoBuf** definition, `ProtobufTrafficFlow.proto`, is attached at the end of this documentation. The following sections explain the different messages used in the data structure in detail.

All fields are marked as optional as recommended by Google but the output will always contain flow information including a location.

Additionally, only messages of `TrafficFlow` will be present. The actual type of message will depend on the specific product definition.

## 4.1. TrafficFlowGroup

The traffic flow group is the top level message. It provides meta data covering general information about the flow data and the flow data itself.

### 4.1.1. MetaInformation

The meta information message provides data shared by all traffic flow messages. It contains the creation time, supplier and client information and the format version.

### 4.1.2. SupplierAndClientInfo

The supplier and client information message is used to identify the client and the supplier.

## 4.2. TrafficFlow

The traffic flow message assigns speed information to a location. It may also identify road blockages.

### 4.2.1. Speed

The speed message subsumes all speed and travel time related information. This message consists of

- the current average speed on the affected location (in km/h)
- the current average travel time to pass the affected location or free flow travel time if average speed is 0 km/h (in seconds)
- (optional) a measure of data quality assigned to the value by the supplier. 100% equates to ideal quality a relation between the average speed and the free-flow speed. The value range is from 0 (average speed is 0) to 1 (average speed is equal to free-flow speed)
- (optional) the relative current speed as a percentage of the freeflow speed
- (optional) an indicator of the status of location

### 4.2.2. SpeedWithTimeStamp

A Speed information at a point in a future time. This time value indicates the number of minutes relative to the creation time of this message as established by the `MetaInformation` message.

### 4.2.3. SectionSpeed

The `SectionSpeed` message provides speed information for section of a location. The section is indicated by the offset of the starting point from the start point of the location. Ending offset is either the end of the location or the start of the next section, if the latter is present. It may also identify road blockages.

## 4.3. Location

The location message provides all kind of supported location referencing methods. A location message contains either a TMC description or a binary OpenLR location reference.

## 4.4. ProtoBuf data structure

See [Protobuf versions and samples^ProtobufTrafficFlow\_v4.proto]

## 5. Geographical representation

An important part of the payload is the geographical representation of the road segments. In TomTom Traffic flow bulk feed two geographical representations are supported: TMC and OpenLR. The extended coverage and detailed granularity that TomTom Traffic flow bulk feed offers can only be used with the OpenLR format. As it also prevents interpretation differences, licensing costs and map and version dependencies, we

recommend using OpenLR over TMC. For compatibility with existing platforms we also offer TMC based data.

In DATEX II the data type of the affectedLocation in the schema is a Group of Locations. In TomTom Traffic flow bulk feed the location method used is the LocationByReference, which contains the identifier of a predefinedLocationReference. The predefinedLocationReference is a TMC link identifier or a OpenLR description.

TMC example:

```
<affectedLocation>
  <locationContainedInGroup xsi:type="LocationByReference">
    <predefinedLocationReference>LD01+27442</predefinedLocationReference>
  </locationContainedInGroup>
</affectedLocation>
```

OpenLR example:

```
<affectedLocation>
  <locationContainedInGroup xsi:type="LocationByReference">
    <locationExtension>
      <openlr><binary>CwANeSUjyAENIAFw+aQBAA==</binary></openlr>
    </locationExtension>
    <predefinedLocationReference>OpenLR</predefinedLocationReference>
  </locationContainedInGroup>
</affectedLocation>
```

In ProtoBuf the <location> field is used for this.

## 5.1. TMC

When TMC is used a geographical representation, travel times are measured per TMC link. A TMC link is a part of a road. It consists of a sequence of road elements that starts at a TMC location and ends at a next TMC location in the TMC table, following the specified number of steps, in the specified direction. A TMC link is identified by:

- TMC table (CountryCode, LocationTableNumber, version)
- Primary TMC location, which defines the TMC point location at the end of the stretch.
- Direction, note that the direction is specified as the direction of travel, NOT as the direction of queue growth as in RDS-TMC!
- an indication of the inclusion of internal/external road segments (see below)
- an optional extent (default value is 1)

In general, the set of TMC link identifiers that can be used in an ElaboratedDataPublication is static. That means, for instance, that in subsequent ElaboratedDataPublications the same locations will be used. However, occasionally the set of used TMC link identifiers can be changed, for instance when a new TMC table is introduced.

### 5.1.1. referenceSettings

For a TMC based DATEX2 feed, the referenceSettings contain a locationSetReference. Here a reference is made to a specific version of a TMC location table. The format is A + BB + "v" + C + "." + D, where:

- A = the Country Code (hex format)
- BB = the Location Table Number (2 digits)
- C = the major version number (1 or 2 digits)
- D = the minor version number (1 or 2 digits)

DATEX II example:

```
<referenceSettings>
  <locationSetReference>D01v7.1</locationSetReference>
</referenceSettings>
```

### 5.1.2. TMC link identifier

In TMC based DATEX2, TMC Chain Information is composed of the following parts ACVVLLLLL[xE[E]] :

Part	Description
A	Fixed letter 'L'
C	EBU country code
VV	TMC Location Table code
D	TMC direction of the chain. For possible values: see table below
LLLLL	TMC point location code. If the number is not 5 digits long, zeros will be added up to 5 characters at the beginning of the code.
[xE[E]]	Either empty, when extent = 1, or fixed letter 'x' followed by the extent (1 or 2 digits)

Using the specified TMC table the secondary location (start of the stretch) can be found from the primary location, the direction and the extent. C, VV and LLLLL are corresponding with the TMC Location Reference that is a unique identifier for a TMC location.

In TMC based ProtoBuf, the TMC chain is not having the fixed letter "L" and is therefore composed as follows: CVVDLLLLL[xE[E]]

Part	Description
C	hexadecimal country code as described in IEC 62106
VV	TMC Location Table code
D	TMC direction of the chain. For possible values: see table below
LLLLL	TMC point location code. If the number is not 5 digits long, zeros will be added up to 5 characters at the beginning of the code.
[xE[E]]	Either empty, when extent = 1, or fixed letter 'x' followed by the extent (1 or 2 digits)

### 5.1.3. TMC direction

The possible values for D refer to the direction of travel (i.e. from secondary to primary location).

Value	Direction	Validity
p (lower case)	Positive	Internal + External
n (lower case)	Negative	Internal + External
+	Positive	External
-	Negative	External
P (upper case)	Positive	Internal
N (upper case)	Negative	Internal

Note that the direction used is the direction of travel, so to find the secondary location; do NOT interpret it as the direction of queue growth. If the extent > 1 then the value for <direction> must be 'p' or 'n', including all internal and external road segments. See below for more details on internal and external segments.

### 5.1.4. Internal and external road segments

To know exactly where a TMC link starts and ends, it is required to understand the concept of internal and external road segments. In short, for each TMC point location and each direction, a composition is made of road segments that are part of the TMC location. For instance on a highway junction the roads between the exit and the entry. These are the internal road segments. Furthermore, a path of road segments is constructed that connects TMC locations. The road segments between 2 TMC locations are the external road segments, which are assigned to the TMC location where the path is leading to.

If two consecutive TMC locations have an overlap or a reverse order in the TomTom map, the TMC locations can be combined into one TMC link

with an extent > 1. Typically a TMC link includes both the internal and external road segments of a TMC location. So a TMC link usually starts at the end of the secondary location and ends at the end of the primary location. In exceptional cases the travel time is provided for just the internal road segments or the external road segments. That is indicated in the TMC link identifier.

### 5.1.5. Examples

L817n39984x2

TMC table 817: the Netherlands, primary location 39984, extent 2 (secondary location 39986), negative direction of travel.

LD01+27442

TMC table D01: Germany, location 27442, just the external part in positive direction.

### 5.1.6. Predefined Location Product

In DATEX II a <PredefinedLocationsPublication> is used to predefine a set of locations, which can then be referred to in the <elaboratedDataPublication>. This allows for simplification of the <elaboratedDataPublication>, where typically the elaborated data is published for a fixed set of locations which is then published separately. However, since there is a straightforward relationship between the <predefinedLocationReference> and the corresponding TMC link, the Predefined Location Product is not published.

## 5.2. OpenLR

OpenLR (see <http://www.openlr.org>) is an open source location referencing system for all roads, including areas or roads where no TMC information is available or in products where no TMC information is desired. OpenLR is a dynamic location reference method which allows referencing the complete digital map. The location references make use of the binary format version 3 as described in the OpenLR whitepaper. An open source reference implementation is available for download on the OpenLR website. TomTom also offers support for 3rd parties to implement and test an own implementation.