

Logistyx TREN API User Manual

Last updated: 2021-02-02

Logistyx TREN API User Manual

- Logistyx TREN API Quick Start

 - STEP 1: Create A QueryData Object

 - STEP 2: POST It

 - STEP 3: Receive A ResultData Object

- The QueryData.DatabaseCode Parameter

 - Two Codes

 - One Code

- The Query-Builder Endpoint

 - For Transpax v7 / TME v3

- The TREN Definitive Codes

 - The Bound Codes

 - The Chargeable Factor Codes

 - The Chargeable Weight Code

 - The Division Match Level Code

 - The Goods Classification Code

 - The Holiday Check Code

 - The Incoterm Duty Code

 - The Linehaul Charge Type Code

 - The Origin / Destination Type Code

 - The Rounding Code

 - The Service Level Code

 - The Surcharge Type Code

- TREN Settings

 - bookingCutOffTime [v7; v8]

 - calculateCostPerPackage [v7; v8]

 - calculateCostPerShipment [v7; v8]

 - calculateCostPerStop [v7; v8]

 - currentDateTime [v7; v8]

 - enableExternalCarrierAPIs [v8]

 - enableLogging [v8]

 - engineVersion [v7; v8]

 - ignoreRestrictions [v8]

 - ignoreSurcharges [v8]

 - includeAllIncoterms [v7; v8]

 - returnInvalidResults [v8]

 - returnMoreInfo [v7; v8]

 - roughDateEstimates [v8]

 - StrictDGClassCompliance

 - StrictUNNumberCompliance

 - UseTUOMs

- Deprecated Settings

 - calculateInvalidCostsThreshold [v7]

 - disableForwarderServiceIfIncotermsNotOK [v7]

 - ignoreDivisionForwarderRelationship [v7]

 - ignoreDivisionServiceRelationship [v7]

 - includeAllDeliveryTimes [v8]

Logistyx TREN API Quick Start

To use the TREN (Transport Rate **EN**gine) API follow these steps:

1. Create a QueryData object containing the shipment (or route) details.
2. POST it to <https://your-TREN.logistyx.com/api/v1/engine>
3. Receive a ResultData object containing the list of forwarder (carrier) service alternatives.

STEP 1: Create A QueryData Object

The QueryData object contains the details of the shipment (or route) and other query parameters (most of them optional). An example:

```
{
  "databaseCode": "ACME-TEST",
  "divisionCode": "ACME-DIV-01",
  "targetCurrencyCode": "EUR",
  "shipment": {
    "shipmentId": 0,
    "pickup": {
      "countryCode": "PE",
      "postalCode": "LIMA18",
      "cityName": "Miraflores",
      "requestedDate": "2021-02-02"
    },
    "delivery": {
      "countryCode": "NL",
      "postalCode": "1438AN",
      "cityName": "Oude Meer"
    },
    "packages": [
      {
        "packageId": 1,
        "packageTypeCode": "BOX",
        "weight": 2.3,
        "length": 40,
        "width": 50,
        "height": 60
      },
      {
        "packageId": 2,
        "packageTypeCode": "BOX",
        "weight": 2.3,
        "length": 40,
        "width": 50,
        "height": 60
      }
    ],
    "packageCount": 2,
  }
}
```

```

    "totalWeight": 4.6,
    "totalVolume": 0.24
  },
  "settings":
  {
    "engineVersion": "v8",
    "calculateCostPerPackage": "true",
    "ignoreSurcharges": "true"
  }
}

```

Some notes:

- The QueryData.DatabaseCode parameter determines the database to connect to. Go to the Logistyx Engine Room to find all the database codes. Read "The QueryData.DatabaseCode Parameter" section for more info.
- You can create the QueryData object from scratch or you can use the Query-Builder endpoint to get started. Read "The Query-Builder Endpoint" section for more info.

STEP 2: POST It

POST the QueryData object to <https://your-TREN.logistyx.com/api/v1/engine>

STEP 3: Receive A ResultData Object

The ResultData object contains the list of forwarder (carrier) services that match the requested query parameters. An example:

```

{
  "engineVersion": "8.2018.07.17",
  "resultDataVersion": "1.2018.07.25",
  "forwarderServices":
  [
    {
      "enabled": true,
      "rateCode": "RATE-01",
      "forwarderCode": "DHL",
      "serviceCode": "DHL-X",
      "transitTime": 3,
      "estimatedPickupDate": "2017-12-13",
      "estimatedDeliveryDate": "2017-12-16",
      "deliveryTimeStart": null,
      "deliveryTimeEnd": null,
      "currencyCodeBeforeExchange": "EUR",
      "freightRateBeforeExchange": 120,
      "currencyCode": "EUR",
      "freightRate": 120,
      "totalSurcharge": 0,
      "totalCost": 120
    },
    {
      "enabled": true,
      "rateCode": "RATE-02",

```

```

    "forwarderCode": "UPS",
    "serviceCode": "UPS-X",
    "transitTime": 3,
    "estimatedPickupDate": "2017-12-13",
    "estimatedDeliveryDate": "2017-12-16",
    "deliveryTimeStart": null,
    "deliveryTimeEnd": null,
    "currencyCodeBeforeExchange": "EUR",
    "freightRateBeforeExchange": 140,
    "currencyCode": "EUR",
    "freightRate": 140,
    "totalSurcharge": 0,
    "totalCost": 140
  },
],
"shipmentCostBreakdown":
{
  "listOfCostPerPackage":
  [
    {
      "packageId": 1,
      "rateCode": "RATE-01",
      "partOfFreightRate": 60,
      "partOfTotalSurcharge": 0,
      "partOfTotalCost": 60,
      "percentageOfFreightRate": 50,
      "percentageOfTotalSurcharge": 0,
      "percentageOfTotalCost": 50
    },
    {
      "packageId": 2,
      "rateCode": "RATE-01",
      "partOfFreightRate": 60,
      "partOfTotalSurcharge": 0,
      "partOfTotalCost": 60,
      "percentageOfFreightRate": 50,
      "percentageOfTotalSurcharge": 0,
      "percentageOfTotalCost": 50
    },
    {
      "packageId": 1,
      "rateCode": "RATE-02",
      "partOfFreightRate": 70,
      "partOfTotalSurcharge": 0,
      "partOfTotalCost": 70,
      "percentageOfFreightRate": 50,
      "percentageOfTotalSurcharge": 0,
      "percentageOfTotalCost": 50
    },
    {
      "packageId": 2,
      "rateCode": "RATE-02",
      "partOfFreightRate": 70,
      "partOfTotalSurcharge": 0,
      "partOfTotalCost": 70,
      "percentageOfFreightRate": 50,
      "percentageOfTotalSurcharge": 0,
      "percentageOfTotalCost": 50
    }
  ]
}

```

```
    }  
  ],  
  "listOfSurchargePerPackage": []  
}  
}
```

The QueryData.DatabaseCode Parameter

One TREN v8 database can store multiple (separate) sets of rates. Because of this, the DatabaseCode parameter can be used with 1 or 2 codes.

Two Codes

```
"databaseCode": "TREN-CODE; CUSTOMER-CODE"
```

When two codes separated by a semi-colon are provided, the first should be the TREN code and the second should be the customer code.

- The TREN code determines the DATABASE to connect to.
- The customer code determines the DATASET (set of rates) to use.

Some examples:

```
"databaseCode": "TREN-TEST; ACME-SA"  
"databaseCode": "TREN-TEST; ACME-NA"  
"databaseCode": "TREN-TEST; ACME-WW"  
"databaseCode": "TREN-PROD; ACME-WW"
```

One Code

```
"databaseCode": "CUSTOMER-DATASET-CODE"
```

When only one code is provided, it should be the customer code. The TREN API will connect to its default database (as configured in the web.config or in Azure) and will use the set of rates that correspond to the given customer code.

Some examples:

```
"databaseCode": "ACME-SA"  
"databaseCode": "ACME-NA"  
"databaseCode": "ACME-WW"
```

Please notice, when running the old TREN v7 only the customer code should be provided.

The Query-Builder Endpoint

The Query-Builder endpoint builds a QueryData object from an existing shipment.

For TranspariX v7 / TME v3

GET <https://rate-engine-api.transparix.com/api/v1/engine/query-builder/tx7/{databaseCode}/{shipmentId}>

- The databaseCode identifies the TX7 database to connect to. Go to the Logistyx Engine Room to find all the database codes.
- The shipmentId identifies the shipment to use to build the QueryData object.

The TREN Definitive Codes

The Bound Codes

Definitive Code	Description	Old Code
INBOUND	An inbound shipment.	IB
OUTBOUND	An outbound shipment.	OB
3RDPARTY	A third party shipment.	3P

The Chargeable Factor Codes

Definitive Code	Description	Old Code
DISTANCE	The distance of a route.	idem
LOADING-LENGTH	The loading length of a shipment.	
PACKAGE-COUNT	The number of packages of a shipment.	COLLI_NR ITEM_COUNT PACKAGE_COUNT
PACKAGE-VOLUME	The volume of a package.	
PACKAGE-WEIGHT	The weight of a package.	
SHIPMENT-VOLUME	The volume of a shipment.	VOLUME
SHIPMENT-WEIGHT	The weight of a shipment.	WEIGHT
TIME	The driving time of a route.	idem

The Chargeable Weight Code

Definitive Code	Description	Old Code
ACTUAL-WT	The actual-weight of a shipment.	ACTUAL WT
HEAVIEST-WT	The heaviest value between the actual-weight, the loading-length-weight and the volume-weight.	CHARGEABLE WT
LOADING-LENGTH-WT	The loading-length-weight of a shipment.	LOADING LENGTH WT
VOLUME-WT	The volume-weight of a shipment.	VOLUME WT

The Division Match Level Code

Definitive Code	Description	Old Code
DIVISION	The service matches at the division level.	n/a
DIVISION-GROUP	The service matches at the division-group level.	n/a
DIVISION-COUNTRY	The service matches at the division-country level.	n/a
ALL-DIVISIONS	The service matches all divisions.	n/a

The Goods Classification Code

Definitive Code	Description	Old Code
NONE	No goods-classification.	idem
SENSITIVE	Sensitive goods.	SD
DANGEROUS	Dangerous goods.	DG

The Holiday Check Code

Definitive Code	Description	Old Code
CHECK-EVERY-DAY	Check every day	
CHECK-PICKUP-DELIVERY	Check at pickup and delivery	
DO-NOT-CHECK	Do not check	
CHECK-PICKUP	Check at pickup	
CHECK-DELIVERY	Check at delivery	
CHECK-AT-ORIGIN	Check at origin	
CHECK-AT-DESTINATION	Check at destination	
CHECK-AT-ORIGIN-DESTINATION	Check at origin and destination	

The Incoterm Duty Code

Definitive Code	Description	Old Code
LOAD-TRUCK		
EXPORT-DUTIES		
TRANSPORT-PORT-EXPORT		
UNLOAD-TRUCK-PORT-EXPORT		
LOADING-CHARGES-PORT-EXPORT		
TRANSPORT-PORT-IMPORT		
UNLOADING-CHARGES-PORT-IMPORT		
LOAD-TRUCK-PORT-IMPORT		
TRANSPORT-DESTINATION		
INSURANCE		
IMPORT-CUSTOMS-CLEARANCE		
IMPORT-TAXES		

The Linehaul Charge Type Code

Definitive Code	Description	Old Code
PER-PACKAGE	Per package.	
PER-SHIPMENT	Per shipment.	
PER-WEIGHT-UNIT	Per weight unit.	

The Origin / Destination Type Code

Definitive Code	Description	Old Code
POSTAL-CODE	A postal code (or a range).	
SHIPPING-POINT	A shipping point.	
ZONE	A zone.	

The Rounding Code

Definitive Code	Description	Old Code
TRUNCATE	Truncate to a number of decimals.	
ROUND-HALF-UP	Round any 5+ decimals up. So 1.5 becomes 2.	
ROUND-UP	Round any decimal up. So 1.1 becomes 2.	
ROUND-UP-TO-1/2	Round any decimal up to half. So 1.23 becomes 1.5.	
ROUND-UP-TO-INT	Round any decimal up to integer. So 1.23 becomes 2.	

The Service Level Code

Definitive Code	Description	Old Code
ROUTINE	A routine service.	ROU
PRIORITY	A priority service.	PRI
EMERGENCY	An emergency service.	EME

The Surcharge Type Code

Definitive Code	Description	Old Code
FIXED	The surcharge amount is fixed.	1
PERCENTAGE	The surcharge amount is a percentage of the freight-rate.	2
PER-N-DRIVING-HOURS	The surcharge amount depends on the number of driving hours. There is a surcharge per each N driving hours.	69
PER-N-EXTRA-KG	The surcharge amount depends on the number of extra-kilograms. There is a surcharge per each N extra kilograms.	64
PER-N-EXTRA-KM	The surcharge amount depends on the number of extra-kilometers. There is a surcharge per each N extra kilometers.	68
PER-N-EXTRA-STOPS	The surcharge amount depends on the number of extra-stops. There is a surcharge per each N extra stops.	66
PER-N-EXTRA-PICKUP-STOPS	The surcharge amount depends on the number of extra-PICKUP-stops. There is a surcharge per each N extra PICKUP stops.	
PER-N-EXTRA-DELIVERY-STOPS	The surcharge amount depends on the number of extra-DELIVERY-stops. There is a surcharge per each N extra DELIVERY stops.	
PER-N-KG	The surcharge amount depends on the number of kilograms. There is a surcharge per each N kilograms.	63
PER-N-KM	The surcharge amount depends on the number of kilometers. There is a surcharge per each N kilometers	67
PER-N-PACKAGES	The surcharge amount depends on the number of packages. There is a surcharge per each N packages.	65

TREN Settings

Notes:

- [v7] indicates the setting is compatible with TREN v7.
- [v8] indicates the setting is compatible with TREN v8.
- All settings are optional, use them as required.

bookingCutOffTime [v7; v8]

A time value to be used as default booking cut-off time when a freight-rate does not specify one.

If this value is NOT set, and if the freight-rate does NOT specify a booking cut-off time either, then booking a same-day pickup shipment is possible at any time (because there is no booking cut-off restriction).

Couple examples:

Value	Meaning
"08:00"	If the freight-rate does not specify a booking cut-off time, then the engine will use 08:00. In this case, if the current time is 07:59 booking a same-day pickup is still possible. But if the current time is 08:01, then same-day pickup is no longer possible.
"12:00"	If the freight-rate does not specify a booking cut-off time, then the engine will use 12:00. In this case, if the current time is 11:59 booking a same-day pickup is still possible. But if the current time is 12:01, then same-day pickup is no longer possible.

calculateCostPerPackage [v7; v8]

Note: This is only for shipment-based rate requests (where the QueryData.Shipment object is used).

A boolean value to indicate that the engine should also calculate the cost per package.

Value	Meaning
false [default]	The engine will NOT calculate the cost per package.
true	The engine will calculate the cost per package. This information will be returned in the ResultData.ShipmentCostBreakdown.

calculateCostPerShipment [v7; v8]

Note: This is only for route-based rate requests (where the QueryData.Route object is used).

A boolean value to indicate that the engine should also calculate the cost per shipment.

Value	Meaning
false [default]	The engine will NOT calculate the cost per shipment.
true	The engine will calculate the cost per shipment. This information will be returned in the ResultData.RouteCostBreakdown.ListOfCostPerShipment.

calculateCostPerStop [v7; v8]

Note: This is only for route-based rate requests (where the QueryData.Route object is used).

A boolean value to indicate that the engine should also calculate the cost per stop.

Value	Meaning
false [default]	The engine will NOT calculate the cost per stop.
true	The engine will calculate the cost per stop. This information will be returned in the ResultData.RouteCostBreakdown.ListOfCostPerStop.

currentDateTime [v7; v8]

A date-time value to be used as current local date-time when performing any date or time operations.

If this value is NOT set, then the default value is the current date-time of the server. In other words, if this value is set, then it will override the date-time of the server for all related estimations.

enableExternalCarrierAPIs [v8]

A boolean value to indicate if the engine should try to get some data (like dates or transit time) directly from (supported) external carrier APIs.

enableLogging [v8]

A boolean value to indicate if the engine should return a log of its operations. This is useful during development and troubleshooting, but not recommended in production because it impacts the engine performance.

engineVersion [v7; v8]

This value determines which version of the engine to run.

Value	Meaning
v7	Run TREN v7
v8	Run TREN v8

ignoreRestrictions [v8]

A boolean value to indicate that any restrictions found in the database should be ignored.

This is related to the restrictions uploaded using the restriction-data-sheet via the Engine Room.
This is NOT related to the inherent limitations of each forwarder-service.

For example: A forwarder-service that does NOT support EUROPALLETS, will still NOT be listed for a shipment that has an EUROPALLET. But a restriction that limits the size of EUROPALLETS will be ignored when listing forwarder-services that DO support EUROPALLETS.

Value	Meaning
false [default]	The engine will enforce all service-level, goods-classification or shipment-dimension restrictions.
true	The engine will NOT enforce any service-level, goods-classification or shipment-dimension restrictions.

ignoreSurcharges [v8]

A boolean value to indicate that any surcharges found in the database should be ignored.

Value	Meaning
false [default]	The engine will calculate all surcharges.
true	The engine will NOT calculate any surcharges.

includeAllIncoterms [v7; v8]

A boolean value to indicate that the engine should also return the forwarder-services that do not match the requested Incoterm.

Value	Meaning
false [default]	The engine will NOT return any forwarder-services that do not match the requested Incoterm.
true	The engine will return some forwarder-services that do not match the requested Incoterm.

returnInvalidResults [v8]

A boolean value to indicate that the engine should also return some forwarder-services that do not fully match the requested query parameters.

Value	Meaning
false [default]	The engine will NOT return any forwarder-services that do not fully match the requested query parameters.
true	The engine will return some forwarder-services that partially match the requested query parameters. These forwarder-services will be disabled.

returnMoreInfo [v7; v8]

A boolean value to indicate that the engine should return additional info about each forwarder-service.

Value	Meaning
false [default]	The engine will NOT return the ForwarderService.MoreInfo object.
true	The engine will return the ForwarderService.MoreInfo object.

roughDateEstimates [v8]

A boolean value to indicate that the engine should NOT take in account holidays nor weekends when estimating the pickup and delivery dates.

Value	Meaning
false [default]	The engine will take in account holidays and weekends when estimating the pickup and delivery dates.
true	The engine will NOT take in account holidays nor weekends when estimating the pickup and delivery dates.

StrictDGClassCompliance

If strict-DG-class-compliance is ON (true), then the engine will disable a forwarder service if the dangerous-goods classes are not OK (supported). If it is OFF (false) then the forwarder service will only suffer a match-quality penalty.

StrictUNNumberCompliance

If strict-UN-number-compliance is ON (true), then the engine will disable a forwarder service if the UN numbers are not OK (supported). If it is OFF (false) then the forwarder service will only suffer a match-quality penalty.

UseIUOMs

A boolean value to indicate if the engine should use the International Trade Units Of Measure.

Deprecated Settings

calculateInvalidCostsThreshold [v7]

This setting is deprecated and always 0 in Engine 8.

This means that Engine 8 does NOT calculate the cost of invalid forwarder services.

This is the old description: This value determines the "there are too many forwarder-services so do NOT calculate the cost of the invalid ones" threshold. If this value is NOT set, then the default value is "30".

disableForwarderServiceIfIncotermsIsNotOK [v7]

This setting was replaced by "includeAllIncoterms" in Engine 8.

`disableForwarderServiceIfIncotermsIsNotOK = !includeAllIncoterms`

ignoreDivisionForwarderRelationship [v7]

This setting is deprecated and always TRUE in Engine 8.

To achieve the old FALSE behavior, the calling application should provide the list of acceptable forwarders (`QueryData.ForwarderCodes`).

ignoreDivisionServiceRelationship [v7]

This setting is deprecated and always TRUE in Engine 8.

To achieve the old FALSE behavior, the calling application should provide the list of acceptable services (QueryData.ServiceCodes).

includeAllDeliveryTimes [v8]

A boolean value to indicate that the engine should also return the forwarder-services that do not match the requested delivery time window.

Value	Meaning
false [default]	The engine will NOT return any forwarder-services that do not match the requested delivery time window.
true	The engine will return some forwarder-services that do not match the requested delivery time window.