

Bosch Connected Services API

Your app or service subscribes to specific message types by sending a PUT request to our Connected Services API:

Based on those subscriptions, we send messages from connected cars to HTTP callback endpoints (webhooks) that are provided by your app or service:

| | |
|--------|------------------------------|
| GET | /subscriptions |
| GET | /subscriptions/{callbackUrl} |
| PUT | /subscriptions/{callbackUrl} |
| DELETE | /subscriptions/{callbackUrl} |

| | |
|------|---------------------------------|
| POST | /callbacks/vehicleData |
| POST | /callbacks/tripStart |
| POST | /callbacks/tripEnd |
| POST | /callbacks/accidentNotification |
| POST | /callbacks/preAccident |
| POST | /callbacks/postAccident |
| POST | /callbacks/trip |
| POST | /callbacks/undelivered |

| | |
|------|------------------------------------|
| POST | /rawCallbacks/ccuCrashGpsPosition |
| POST | /rawCallbacks/ccuCrashNotification |
| POST | /rawCallbacks/ccuCrashReport |
| POST | /rawCallbacks/ccuDrivingProfile |
| POST | /rawCallbacks/ccuDtcReport |
| POST | /rawCallbacks/ccuEndOfTrip |
| POST | /rawCallbacks/ccuFuelReport |
| POST | /rawCallbacks/ ... |

REST, HTTP, JSON
Swagger, Webhook

Examples for our message types

```
tripMessage {
  id*          string
              Unique ID of trip.

  deviceId*    string
              Unique ID of the device that recorded the trip data.

  tripVersion  integer($int32)
              Counter (starting with 1) incremented for each new version of the trip.

  startDateTime* string($date-time)
              Date and time (defined by date-time in RFC3339) on the start of the trip.

  endDateTime  string($date-time)
              Date and time (defined by date-time in RFC3339) on the end of the trip.

  startMileage integer($int32)
              Vehicle mileage value at the start of this trip in meters.

  endMileage   integer($int32)
              Vehicle mileage value at the end of this trip in meters.

  distance     integer($int32)
              Distance travelled on the trip in meters.

  finished     boolean
              False if trip is still ongoing, i.e. not finished yet.

  positions    {
              GPS positions of the trip route.

  geoPosition {
    accuracy      number($float)
                  Accuracy of GPS position

    latitude      integer($int64)
                  Specifies the north-south position of a point in
                  milliarcseconds

    longitude     integer($int64)
                  Specifies the east-west position of a point in
                  milliarcseconds

    altitude      integer($int64)
                  Specifies the height above sea level in meters

    direction     integer($int32)
                  Direction of travel in degrees; 0 = north, 90 =
                  east

    recordedDateTime string($date-time)
                  Date and time (defined by date-time in RFC3339)
                  when this position was recorded.
```

```
tripStartMessage {
  messageId      string
                  Unique ID of the message

  deviceId        string
                  Unique ID of the device

  vehicleId      string
                  Unique ID of the vehicle

  tenantId       string
                  Unique ID of the tenant

  tripId         string
                  Unique ID of the trip

  messageVersion string
                  Version of message format, following semantic versioning

  firmwareVersion string
                  Version of firmware, following semantic versioning

  timestamp      integer($int64)
                  Timestamp of event
```

...

www.boschconnectedvehicle.com