1. Overview

The BlueFire API for Android development is available on GitHub at the following location:

- Android Studio - https://github.com/BlueFire-LLC/BlueFire-API-for-Android-Studio

The GitHub repository consists of an API Demo app and the libraries needed to perform custom application development. The BlueFire API library is named bluefire-api-vxx.x.jar where vxx.x is the current version (i.e. v24.0). All the libraries in the lib folder are required for the API Demo.

2. General Methods and Properties

2.1. BlueFire (Constructor)

Parameters:
Context: Android app context
Handler: Android handler for handling events

Description: Constructor for the API.

2.2. APIVersion

Type: Integer
Data: API version number

Description: Can be used to verify current API version.

2.3. AndroidVersion

Type: Byte Array
Data: Android version number

Description: Can be used to verify current Android version.

2.4. Dispose

Parameters: None
Context:

Description: Disconnects the API from the Adapter and disposes the API.

3. Methods and Properties before Adapter Connection

3.1. UseBLE

Type: Boolean
Data: Connect using BLE (Bluetooth Low Energy).

Description: Instructs the API to connect to a BLE Adapter. If UseBLE and UseBT21 are either both set or both not set, the API will automatically determine the type of Adapter and set the appropriate property (UseBLE or UseBT21).

Note: There are limitations to the BLE Adapter on Android. See the README on GitHub for more information.

3.2. *UseBT21*

Type: Boolean
Data: Connect using Bluetooth 2.1 (Bluetooth Classic).

Description: Instructs the API to connect to a Bluetooth Classis Adapter. If UseBT21 and UseBLE are either both set or both not set, the API will automatically determine the type of Adapter and set the appropriate property (UseBT21 or UseBLE).

Note: Bluetooth Classic uses Com sockets and they can block for a considerably amount of time depending on the OEM device. It is therefore recommended that you adjust the MaxConnectAttempts, MaxReconnectAttempts, and the DiscoveryTimeout to compensate for this duration.

3.3. *SetConnectToLastAdapter*

Parameters:
ConnectToLastAdapter (Boolean – optional, default is false)

Description: Setting this true will cause the API to only connect to an Adapter that has its Bluetooth Device Id equal to the last connect id that was set using the SetAdapterId method.

3.4. *SetAdvertisementTimeout*

Parameters:
AdvertisementTimeout – in milliseconds, default is 5000 (5 seconds).

Description: This is the time in milliseconds that the API will wait for a BlueFire BLE adapter to be found during its advertisement scan. The default is 5 seconds. Note, normally this is very quick but if there are many BLE devices in the general vicinity, finding a BlueFire adapter could take longer.

3.5. *SetDiscoveryTimeout*

Parameters:
DiscoveryTimeout – in milliseconds, default is 30000 (30 seconds).

Description: This is the time in milliseconds that the API should wait for the Bluetooth Discovery process to find an Adapter. The default is 30 seconds. The Discovery process is a blocking process so care should be taken when setting this value. Note, the API will look for paired Adapters first before initiating the Discovery process.

3.6. *SetUseInsecureConnection*

Parameters:
UseInsecureConnection (Boolean – optional, default is false)

Description: Set this true to force the API to connect to the Adapter using an insecure Bluetooth connection. Some Android devices will not connect to the Adapter using a secured Bluetooth connection but only with an insecure connection. Note that there is no security risk with using an insecure connection.

Note: This is only applicable for Bluetooth 2.1 (Classic) adapters.

3.7. *SetMaxConnectAttempts*

Parameters:

MaxConnectAttempts (Integer)

Description: This is the maximum number of retries the API should attempt during an Adapter connection. Many Android devices require multiple connection attempts before successfully connecting. However each retry can take several seconds so this value should be set carefully and accurately. The default is 5 attempts.

3.8. *SetMaxReconnectAttempts*

Parameters:

MaxReconnectAttempts (Integer)

Description: This is the maximum number of Adapter reconnection attempts the API should try before giving up. Many Android devices require multiple reconnection attempts before successfully connecting. However each attempt can take several seconds so this value should be set carefully and accurately. The default is 5 attempts.

3.9. *SetBluetoothRecycleAttempt*

Parameters:

BluetoothRecycleAttempt (Integer)

Description: This is the connection or reconnect attempt that Bluetooth will be recycled (turned off and back on). The default is to recycle on the 2nd attempt.

3.10. *SetBleDisconnectWaitTime*

Parameters:

BleDisconnectWaitTime - in milliseconds, default is 2000 (2 seconds

Description: This is the amount of time that the API waits for Android to disconnect and close the GATT connection after an API Disconnect.

Note: Note, in order for Bluetooth LE to release the connection to the adapter and allow reconnects or subsequent connects, it must be completely closed. Unfortunately Android does not have a way to detect this other than waiting a set amount of time after disconnecting from the adapter. This wait time can vary with the Android version and the make and model of the mobile device. The default is 2 seconds. If you experience numerous unable to connect and BlueFire LE fails to show up under Bluetooth settings, try increasing this value.

3.11. *SetIgnoreJ1939*

Parameters:

IgnoreJ1939 (Boolean)

Description: Sets whether the J1939 Databus should be ignored. This must be called before connecting to the Adapter. The default is false (connect to J1939).

Note: Setting this false will set IgnoreOBD2 true (Firmware 3.15+).

3.12. SetIgnoreJ1708

Parameters:

IgnoreJ1708 (Boolean)

Description: Sets whether the J1708 Databus should be ignored. If is recommended that if J1708/J1587 data is not required this should be set to ignore. This must be called before connecting to the Adapter. The default is true (do not connect to J1708).

Note: Setting this false will set IgnoreOBD2 true (Firmware 3.15+).

3.13. SetIgnoreOBD2

Parameters:

IgnoreOBD2 (Boolean)
CANSetting (OBD2.CANSettings)

Description: Sets whether the OBD2 Databus should be ignored. This must be called before connecting to the Adapter. The default is true (do not connect to OBD2).

Note: Setting this false will set IgnoreJ1939 and IgnoreJ1708 true. This property is only available with Firmware 3.15+.

Warning: Changing the CANSetting from the default (NoEID_500K) may cause ECM faults.

3.14. SetKeepJ1708PortOpen

Parameters:

KeepJ1708PortOpen (Boolean)

Description: Instructs the Adapter to keep the J1708 Com port open. This must be called before connecting to the Adapter.

Note: This can be used for vehicles that do not adhere to the J1708 spec and raise a fault when the Adapter is connected.

3.15. SetForce500kBus

Parameters:

Force500kBus (Boolean)

Description: Instructs the Adapter to only connect to the 500K CAN Bus. This must be called before connecting to the Adapter.

Note: This can improve CAN Bus connection time since the Adapter will always attempt to connect to the 250K CAN bus first before trying the 500K Bus.

3.16. SetMinInterval

Type: Integer
Data: The minimum interval (in milliseconds) for retrieving data from the Adapter.

Description: Sets the Adapter's minimum data transmit interval. This will reduce the Bluetooth data traffic and may help with Adapter connection issues. This is set by default to 500 milliseconds (.5 seconds) for BLE Adapters due to the limitations of Android BLE.

3.17. SetPerformanceInterval

Type: Integer
Data: The performance interval (in milliseconds) for use by the SetPerformanceMode.

Description: Sets the Adapter's performance interval. The default is 500 milliseconds (.5 seconds).

3.18. SetPerformanceModeOn

Parameters:
SetPerformanceModeOn (Boolean)

Description: Set this true to turn on the Adapter's performance mode and false to turn it off. This improves the performance of slower PGNs (Brake Pressures, Odometer, Temps, etc.) by interrupting faster PGNs (RPM, Speed, Fuel Economy, etc.). See the PerformanceInterval property for the interrupt interval.

Note: Performance Mode is only available on Adapters with Firmware 3.8+.

3.19. SendAllPackets

Parameters:
SendAllPackets (Boolean)

Description: Set this true to instruct the Adapter to send all Vehicle VIN and Vehicle Id packets all at once instead of using the RTS conversation mode. This may improve the retrieval of VIN and Vehicle Id data (Make, Model, Serial No, etc).

Note: This property is only available with Firmware 3.15+.

3.20. OptimizeDataRetrieval

Parameters: None
Type: Boolean
Data: The current OptimizeDataRetrieval setting.

Description: This indicates whether optimize data retrieval is being used.

3.21. SetOptimizeDataRetrieval

Parameters:
Optimize Data Retrieval (Boolean – true to optimize)

Description: Turns on or off the optimize data retrieval feature. This is only applicable if both J1939 and J1708 are in use (ie. not ignored) in which case if data is retrieved from J1939, the request for data from J1708 is removed. This will improve performance and stability.

3.22. SetDeviceId

Parameters:
DeviceId (String)

Description: Sets the Device Id that the API will use to compare with the connecting Adapter to verify a valid connection. This check only occurs if SecureDevice is set to true. This must be called before connecting to the Adapter.

Note: The Device Id is unique to the mobile device (phone, tablet, etc.) and is created when the API is instantiated for the first time by the App.

3.23. SetAdapterId

Parameters:
AdapterId (String)

Description: Sets the Adapter Id that the API will use to compare with the connecting Adapter to verify a valid connection. This check only occurs if ConnectToLastAdapter or SecureAdapter is set to true. This must be called before connecting to the Adapter.

3.24. SetSecurity

Parameters:
SecureDevice (Boolean)
SecureAdapter (Boolean)
UserName (String)
Password (String)

Description: Sets the security parameters to be used for authentication with the Adapter. If security authentication is not being used this method is not necessary. All authentication uses AES encryption.

SecureDevice secures the App on the particular device with an Adapter. One device can be secured to many Adapters. This is a one to many relationship.

SecureAdapter secures the App to the Adapter and will not allow any other App to connect to the Adapter. Likewise the App will not be able to connect to any other Adapter. This is a one to one relationship.

UserName and Password secures the App to an Adapter. An App can be secured to many Adapters and many Adapters can be secured to an App on other devices. This is a many to many relationship. The UserName and Password are case sensitive.

3.25. DisconnectedReboot

Type: Boolean
Data: The DisconnectedRebootInterval (in milliseconds) for use by the SetDisconnectedReboot.

Description: Instruct the Adapter to reboot at a set interval when not connected to the API.

Note: With Firmware 3.12+ the interval is fixed at one hour. With Firmware 3.15+ the interval is set according to the DisconnectedRebootInterval.

3.26. Connect

Parameters: None

Description: Initiate a connection attempt to the Adapter. This is a blocking call so it is best to call this from a thread and handle the events generated from the connection attempt.

4. Methods and Properties after Adapter Connection

4.1. IsCompatible

Type: Boolean
Data: Indicates if the Adapter is compatible with the API.

Description: The Adapter must have Firmware Version 3.7 or greater.

4.2. IsConnected

Type: Boolean
Data: Indicates that the API is connected to the Adapter.

Description: See the API Demo for best use for this.

4.3. ConnectAttempt

Type: Integer
Data: The current connection attempt.

Description: This will be between 1 and MaxConnectAttempts.

4.4. IsReconnecting

Type: Boolean
Data: Indicates that the API has lost connection to the Adapter and is attempting to reconnect.

Description: See the API Demo for best use for this.

4.5. ReconnectAttempt

Type: Integer
Data: The current re-connection attempt.

Description: This will be between 1 and MaxReconnectAttempts.

4.6. IsCANAvailable

Type: Boolean
Data: Indicates if the J1939 or OBD2 CAN bus is available.

Description: In most vehicles the CAN bus is detectable only when the key is on.

Note: This checks for the existence of PGN 65262 on the CAN bus. Use the IsOBD2 property to determine if the J1939 or OBD2 CAN bus is available.

4.7. IsJ1708Available

Type: Boolean
Data: Indicates if the J1708 bus is available.

Description: This check for the existence of PID 1 on the J1708 data bus. In most vehicles the J1708 bus is detectable only when the key is on.

4.8. IgnoreJ1939

Type: Boolean
Data: Indicates if the J1939 Databus is being ignored.

Description: Indicates whether the J1939 Databus is being ignored. The default is false (connect to J1939).

Note: If the HardwareType is a 6-pin Adapter this will automatically be set true.

4.9. IgnoreJ1708

Type: Boolean
Data: Indicates if the J1708 Databus is being ignored.

Description: Indicates whether the J1708 Databus is being ignored. The default is true (do not connect to J1708).

Note: If the HardwareType is a 6-pin Adapter this will automatically be set false.

4.10. IgnoreOBD2

Type: Boolean
Data: Indicates if the OBD2 Databus is being ignored.

Description: Indicates whether the OBD2 Databus is being ignored. The default is true (do not connect to OBD2).

4.11. IsOBD2

Type: Boolean
Data: Indicates if data is being retrieved from the OBD2 Databus.

Description: This can be used to differentiate between the J1939 CAN bus data or the OBD2 CAN bus is available (see IsCANAvailable).

4.12. GetKeyState

Parameters: None

Description: Retrieves the IsCANAvailable and IsJ1708Available properties from the Adapter.

Note: The IsCANAvailable and IsJ1708Available properties are automatically updated by the Adapter when the states change so this method is just a double check of the available states.

4.13. IsKeyOn

Type: Boolean
Data: Indicates if the vehicle's ignition key is on (true) or off (false).

Description: This uses the IsCANAvailable and/or IsJ1708Available properties to determine if state of the ignition key (on or off)

Note: When the key is turned off (state changes on to off), Vehicle RPM, Speed, PctLoad, PctTorque, and DrvPctTorque are set to 0.

4.14. IsKeyOff

Type: Boolean
Data: Indicates if the vehicle's ignition key is off (true) or on (false).

Description: This uses IsKeyOn to determine the ignition key state.

4.15. ConnectionState

Type: ConnectionStates
Data: The state of connecting to the Adapter.

Description: This encompasses the Bluetooth connection and the Adapter connection.

4.16. ConnectionMessage

Type: String
Data: A message from the API or the Adapter.

Description: This can be an informational or an error message. It should be logged for later analysis.

4.17. DeviceId

Type: String
Data: The App's Device (phone, tablet, etc.) Id.

Description: This is an API generated Id that is unique to the App installed on a particular device;

4.18. AdapterId

Type: String
Data: The Adapter's unique Id.

Description: This is the Bluetooth MAC Id formatted as hexadecimal "nn:nn:nn:nn:nn:nn";

4.19. FirmwareVersion

Type: String
Data: The Adapter's Firmware Version

Description: Format is "nn.nn".

4.20. HardwareVersion

Type: String
Data: The Adapter's Hardware Version

Description: Format is "nn.nn".

4.21. HardwareType

Type: HardwareTypes
Data: The Hardware Type of the Adapter

Description: This indicates if the adapter is a 6-pin or 9-pin adapter and if it supports J1939 and J1708.

4.22. CANBusSpeed

Type: CANBusSpeeds
Data: The CAN bus speed the Adapter is using.

Description: This is the CAN bus speed the Adapter detected and is using.

4.23. SerialNo

Type: String
Data: The Adapter's Serial Number.

Description: The serial number assigned at manufacturing. Note, if the adapter firmware is flashed over the top of an older incompatible firmware, the serial number will be cleared and the API will generate a new serial number.

4.24. SleepMode

Type: SleepModes
Data: Sleep mode of the Adapter

Description: The current sleep mode of the Adapter.

4.25. SetSleepMode

Parameters:
SleepMode (SleepModes)

Description: Sets the Adapter's Sleep Mode. The Adapter can be set for NoSleep or WakeOnApp. When WakeOnApp is used, the Adapter will go into sleep mode two minutes after no activity and will wake up when the API initiates a connection attempt.

4.26. LedBrightness

Type: Integer
Data: The current brightness setting for the Adapter LEDs.

Description: This will be in the range of 5 to 100.

4.27. SetLedBrightness

Parameters:
LED Brightness Level (Integer - valid values are 5 to 100)

Description: Sets the brightness level of the Adapter's LEDs. Note, the brightness level is not allowed to be less than 5 due to the importance of the state of the LEDs.

4.28. SetPerformanceInterval

Type: Integer
Data: The performance interval (in milliseconds) for use by the SetPerformanceMode.

Description: Sets the Adapter's performance interval. The default is 500 milliseconds (.5 seconds).

4.29. SetPerformanceModeOn

Parameters:
SetPerformanceModeOn (Boolean)

Description: Set this true to turn on the Adapter's performance mode and false to turn it off. This improves the performance of slower PGNs (Brake Pressures, Odometer, Temps, etc.) by interrupting faster PGNs (RPM, Speed, Fuel Economy, etc.). See the PerformanceInterval property for the interrupt interval. Note, Performance Mode is only available on Adapters with Firmware 3.8+.

4.30. SetTime

Parameters: None

Description: Sets the Adapters clock to the current App's UTC date and time. This is primarily used for ELD recording.

4.31. SetHeartbeatOn

Type: Boolean
Data: The Adapter's heartbeat setting, on (true) or off (false). The default is On.

Description: Setting this false will turn off the Adapter heartbeat and cause the Adapter to ignore its heartbeat timeout. Use this with caution as it will cause the Adapter to stay connected indefinitely if the App force closes.

4.32. HeartbeatCount

Type: Integer
Data: A numerical count of the heartbeat received from the Adapter.

Description: Can be used to visually show the connection to the Adapter. If the heartbeat stops the API will attempt to reconnect to the Adapter.

4.33. SetNotificationsOn

Type: Boolean
Data: The Adapter's notification setting, on (true) or off (false). The default is Off.

Description: Setting this true will cause the Adapter to send all notifications to the App.

4.34. GetMessages

Parameters: None

Description: Gets any Adapter messages.

4.35. ResetAdapter

Parameters: None

Description: Resets the Adapter to factory settings. The API will disconnect from the Adapter and start the factory reset which will take approximately 45 seconds. A re-connection should not be attempted until the factory reset is complete.

For Adapters with Firmware 3.17 the Adapter will alternately blink red, green, and blue when the reset is complete. The Adapter must be unplugged and plugged back in before reconnecting to it.

For Adapters with Firmware 3.18 the Adapter will blink blue when the reset is complete and can be reconnected right away.

To manually reset the Adapter follow the instructions in the Appendix.

4.36. RebootAdapter

Parameters: None

Description: Initiates a reboot of the Adapter. When the Adapter reboots, the API will detect a disconnection and will attempt to reconnect.

4.37. Disconnect

Parameters:
WaitForDisconnect (Boolean - optional, default is false)

Description: Disconnects the API from the Adapter and initiates a reboot of the Adapter. If the WaitForDisconnect parameter is true, the call will block until the Adapter has disconnected.

5. Vehicle Data Methods and Properties after Adapter Connection

5.1. IsVehicleDataChanged

Type: Boolean
Data: Indicates that vehicle data retrieved from PGNs has changed.

Description: This is set true whenever PGN data that is being monitored is received by the API. The default for monitoring PGNs is to only send data when PGN data has changed.

5.2. GetEngineVIN

Parameters:
Retrieval Method (RetrievalMethods - optional, default is OnInterval).
Interval (Integer – in milliseconds, optional, default is MinInterval).

Returns:
False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing).

Vehicle Data:
Engine VIN

Description: Retrieves the engine VIN from the Adapter.

5.3. GetEngineId

Parameters:
Retrieval Method (RetrievalMethods - optional, default is OnInterval).
Interval (Integer – in milliseconds, optional, default is MinInterval).

Returns:
False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing).

Vehicle Data:
Make

Model
Serial Number
Unit Number

Description: Retrieves the engine identification from the Adapter.

5.4. StopRetrievingEngineVIN

Parameters: None

Description: Stops the retrieval of the engine VIN from the Adapter. This should be called after the VIN has been retrieved.

5.5. StopRetrievingEngineId

Parameters: None

Description: Stops the retrieval of the engine component Id from the Adapter. This should be called after the component Id has been retrieved.

5.6. GetVehicleVIN

Parameters:

Retrieval Method (RetrievalMethods - optional, default is OnInterval).
Interval (Integer – in milliseconds, optional, default is MinInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing).

Vehicle Data:

VIN

Description: Retrieves the OEM VIN from the Adapter.

5.7. GetVehicleId

Parameters:

Retrieval Method (RetrievalMethods - optional, default is OnInterval).
Interval (Integer – in milliseconds, optional, default is MinInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing).

Vehicle Data:

Make
Model
Serial Number
Unit Number

Description: Retrieves the OEM component identification from the Adapter.

5.8. *StopRetrievingVehicleVIN*

Parameters: None

Description: Stops the retrieval of the vehicle VIN from the Adapter. This should be called after the VIN has been retrieved.

5.9. *StopRetrievingVehicleId*

Parameters: None

Description: Stops the retrieval of the vehicle component Id from the Adapter. This should be called after the component Id has been retrieved.

5.10. *GetEngineData1*

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

RPM
Percent Torque
Driver Torque
Torque Mode

Description: Retrieves engine data from the Adapter.

5.11. *GetEngineData2*

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Percent Load
Accelerator Pedal Position

Description: Retrieves engine data from the Adapter.

5.12. *GetEngineData3*

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Max Speed
Vehicle Speed
Brake Switch
Clutch Switch
Park Brake Switch
Cruise Control Switches

Description: Retrieves engine data from the Adapter.

5.13. GetDistance

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Engine Distance – high resolution if available otherwise low resolution
High Resolution Distance,
Low Resolution Distance,

OEM Odometer – high resolution if available otherwise low resolution
High Resolution Distance,
Low Resolution Distance,

Description: Retrieves Engine Distance (high resolution and/or low resolution) and Odometer from the Adapter. Note, Engine Distance is retrieved from the Engine ECM and Odometer is retrieved from the OEM ECM (if available).

5.14. GetOdometer

Same as GetDistance.

Note: 2017+ Volvo and 2018 Kenworth and Peterbilt vehicles no longer transmit Odometer.

5.15. GetTemps

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Oil Temp
Coolant Temp
Intake Manifold Temp

Transmission Temp

Description: Retrieves temperature data from the Adapter.

5.16. GetPressures

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Oil Pressure
Coolant Pressure
Intake Manifold (Boost) Pressure

Description: Retrieves engine temperatures from the Adapter.

5.17. GetCoolantLevel

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Coolant Level

Description: Retrieves the engine coolant level from the Adapter.

5.18. GetFuelData

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Fuel Rate
Throttle Position
Instant Fuel Economy
Average Fuel Economy
Total Low Res Fuel Used
Total High Res Fuel Used
Total Idle Fuel Used
Primary Fuel Level
Secondary Fuel Level

Description: Retrieves engine fuel data from the Adapter.

Note: The 2018 Freightliner vehicles no longer return Fuel Used.

5.19. GetEngineHours

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Total Idle Hours
Total Engine Hours

Description: Retrieves engine hours from the Adapter.

5.20. GetBrakeData

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Brake Application Pressure
Brake Primary Pressure
Brake Secondary Pressure

Description: Retrieves brake data from the Adapter.

5.21. GetTransmissionGears

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Current Gear
Selected Gear

Description: Retrieves the transmission gears from the Adapter. This will only be available if the transmission ECM provides the data.

5.22. GetBatteryVoltage

Parameters:

RetrievalMethod (RetrievalMethods - see Enums, optional, default is OnChange)
Interval (Integer – in milliseconds, optional, required for RetrievalMethod OnInterval).

Returns:

False if unable to retrieve synchronized data, otherwise the following data will be returned in the Vehicle Data class (asynchronously if not synchronizing):

Battery Potential (Voltage)

Description:

5.23. GetELDData

Parameters: None

Returns:

RPM
Vehicle Speed
Engine Distance
Vehicle Odometer
Total Hours

Description: Retrieves ELD data using preset optimum settings from the Adapter. The settings are:

RPM – OnChange, one second interval
Speed – OnChange, 500 ms interval
Distance/Odometer – OnChange, one second interval
Total Hours - OnChange, one second interval

5.24. GetFaults

Parameters:

J1939 Source (Integer – optional, default is Engine).
J1587 MID (Integer – optional, default is Engine).

Returns:

SPN
FMI
Occurrence
Conversion
IsJ1587Fault

Description: Retrieves faults from the Adapter. Use the Vehicle class methods to retrieve the Fault data. This is an asynchronous method and faults will be updated when they occur or when they are reset.

Note: This clears the CAN Filter so it must be called before any other data requests.

5.25. ResetFaults

Parameters: None

Description: Resets any active faults. Note, the API sends the appropriate Fault Reset command to all ECUs. It is up to the ECUs to accept and process the Reset command.

5.26. StopDataRetrieval

Parameters: None

Description: Stops all data retrieval from the Adapter.

6. J1939 AND J1708 METHODS AND PROPERTIES

6.1. PGNData

Type: Class
Data:

PGN (Integer)
Source (Integer)
Data (Byte [8])

Description: Retrieves the PGN data that is returned by the Adapter from the StartMonitoringPGN, RequestPGN, and SendPGN methods.

6.2. StartMonitoringPGN

Parameters:

Source (Integer)
PGN (Integer)
Interval (Integer) – in milliseconds, optional, default is 0 (on data change).
IsOnRequest (Boolean) – optional, default is false.
IsBAMRTS (Boolean) – optional, default is false.

Description: Starts monitoring a PGN. Data will be returned in the PGNData class and is based on the Interval. If IsOnRequest is true, the PGN will be sent to the CAN Bus as an on-request PGN. If IsBAMRTS is true, the PGN will be monitored as a multi-packet BAM or RTS PGN. The IsOnRequest and IsBAMRTS must match the SAE specification of PGN in order for data to be retrieved correctly.

Note: The maximum number of concurrent monitoring PGNS is 20. Use the StopMonitoringPGN method to remove a PGN from this count.

6.3. StopMonitoringPGN

Parameters:

Source (Integer)
PGN (Integer)
IsBAMRTS (Boolean) – optional, default is false.

Description: Stops monitoring a PGN. This will remove the PGN from the maximum number of monitoring PGNs (20).

6.4. RequestPGN

Parameters:

Source (Integer)
PGN (Integer)
IsOnRequestPGN (Boolean – optional, default is false)

IsBAMRTS (Boolean) – optional, default is false.

Description: Request PGN data once only. If IsOnRequestPGN is false, the PGN will be monitored and will return the first data received. If IsOnRequestPGN is true, the PGN will be sent to the CAN Bus as an on-request PGN. If IsBAMRTS is true, the PGN will be monitored as a multi-packet BAM or RTS PGN. The IsOnRequest and IsBAMRTS must match the SAE specification of PGN in order for data to be retrieved correctly.

6.5. SendPGN

Parameters:

Source (Integer - optional, default is 43)
PGN (Integer)
Interval (Integer)
Priority (Byte - optional, default is 6)
PGN Data (Byte Array, size is 8)

Description: Send a non-standard API PGN to the Adapter. Any response can be retrieved with the PGNData property.

6.6. PIDData

Type: Class
Data:

MID (Integer)
PID (Integer)
Data (Byte [8])

Description: Retrieves the MID data that is returned by the Adapter from the MonitorPID and RequestPID methods.

6.7. RequestPID

Parameters:

MID (Integer)
PID (Integer)
IsOnRequestPGN (Boolean – optional, default is false)

Description: Request J1587 PID data once only. If IsOnRequestPGN is false, the PID will be monitored and will return the first data received. If IsOnRequestPGN is true, the PID will be sent to the J1708 Bus as an on-request PID.

6.8. MonitorPID

Parameters:

MID (Integer)
PID (Integer)
Interval (Integer – in milliseconds, optional, default is 0 (on data change))
IsOnRequestPGN (Boolean – optional, default is false)

Description: Monitors a J1587 PID. Data will be returned based on the Interval. If IsOnRequestPGN is true, the PID will be sent to the J1708 Bus as an on-request PID.

6.9. StopMonitoringPID

Parameters:

MID (Integer)
PID (Integer)

Description: Stops monitoring a J1587 PID.

7. ENUMS

7.1. HardwareTypes

HW_1_1

First version 9 pin J1939 only

HW_6_Pin

6 pin J1939 and J1708

HW_9_Pin

9 pin J1939 and J1708

7.2. CANBusSpeeds

NA	Not assigned
K250	250K Bus
K500	500K Bus
Error	Both 250K and 500K

7.3. CANSettings (OBD2)

NoEID_500K	Standard Frame (11 bit) 500K (default)
NoEID_250K	Standard Frame (11 bit) 250K
EID_500K	Extended Frame (29 bit) 500K
EID_250K	Extended Frame (29 bit) 250K

7.4. RetrievalMethods

OnChange (default)

Data will be retrieved when it changes. Use this with caution with Android as BLE may become unstable if the data rate is high.

OnInterval

Data will be retrieved on the specified interval only if it changes. If the interval is less than MinInterval, MinInterval will be used.

Synchronized

Data will be retrieved immediately. This will force a blocking call. Timeout occurs after the SyncTimeout value expires. This can be changed with the SetSyncTimeout method.

7.5. ConnectionStates

The ConnectionState and ConnectionMessage are sent from the API to the App via an Event Handler. The ConnectionMessage will be populated according to the ConnectionState. It is imperative that the Event Handler be allowed to process the event without interruption otherwise events can be missed and the App will not be notified of critical events.

Initializing
Initialized

Occurs at the beginning of the connection.

Discovering

Occurs if Bluetooth Discovery is invoked.

Connecting

Occurs when the App is connecting to Bluetooth.

Connected

Occurs when the App is connected to Bluetooth.

IsReady

Occurs after connecting and authenticating to the Adapter and the Adapter settings have been retrieved. These include PerformanceMode, SleepMode, LedBrightness, IgnoreJ1939, IgnoreJ1708, HardwareType and any Adapter Messages.

NotAuthenticated

Occurs after connecting to the Adapter and the API failed to authenticate the Adapter with the App security.

Disconnecting

Disconnected

Occurs when the App or the API is disconnecting from the Adapter.

NotConnected

Occurs when the API is not connected to the Adapter.

Reconnecting

Reconnected

Occurs when the App or the API is reconnected to the Adapter.

NotReconnected

Occurs when the API is reconnecting to the Adapter after a loss of connection.

CANStarting

Occurs when the Adapter is starting the CAN connection.

J1708Restarting

Occurs when the Adapter is restarting the J1708 connection.

ELDConnected

Occurs when the API receives ELD startup data from the adapter after the ELD Connect method is called.

Heartbeat

Occurs when the API receives a heartbeat from the Adapter.

DataAvailable

Occurs when there is Vehicle data available for the App to process.

Notification

Occurs when there is a notification from the API. The ConnectionMessage will contain the notification text.

AdapterMessage

Occurs when there is a message from the Adapter. The ConnectionMessage will contain the message text.

CANFilterFull

Occurs when too many data requests have been sent to the Adapter. This applies specifically to J1939 PGNs.

DataError

Occurs when a data error is detected by the API. The ConnectionMessage will contain the error text. The API will attempt a reconnection.

DataTimeout

Occurs when the data connection between the API and the Adapter is lost. The API will attempt a reconnection.

BluetoothTimeout

Occurs when the API cannot connect to Bluetooth.

AdapterTimeout

Occurs when the API cannot connect to the Adapter.

AdapterReboot

Occurs when the Adapter is rebooting for a specific reason. The ConnectionMessage will contain the reboot reason. The API will attempt a reconnection.

SystemError

Occurs when the API encounters a code exception. The ConnectionMessage will contain the exception information.

8. Vehicle Data

8.1. *RPM*

Type: Integer
Data: Engine RPM

8.2. *Speed*

Type: Float
Data: Vehicle Road Speed (metric, kph)

8.3. *AccelPedal*

Type: Float
Data: Accelerator Pedal Position (0-100%)

8.4. *ThrottlePos*

Type: Float
Data: Throttle Position (0-100%)

8.5. *MaxSpeed*

Type: Integer
Data: Maximum Set Speed (kph)

8.6. *HiResMaxSpeed*

Type: Float
Data: High Resolution Maximum Set Speed (kph)

8.7. *Distance*

Type: Float
Data: High/Low Resolution Engine Distance (meters)

8.8. *HiResDistance*

Type: Float
Data: High Resolution Engine Distance (meters)

8.9. *LoResDistance*

Type: Float
Data: Low Resolution Engine Distance (kilometers)

8.10. Odometer

Type: Float
Data: High/Low Resolution OEM Distance (meters)

8.11. HiResOdometer

Type: Float
Data: High Resolution OEM Distance (meters)

8.12. LoResOdometer

Type: Float
Data: Low Resolution OEM Distance (kilometers)

8.13. TotalHours

Type: Float
Data: Total Engine Hours

8.14. IdleHours

Type: Float
Data: Total Engine Idle Hours

8.15. PctLoad

Type: Integer
Data: Percent Load

8.16. PctTorque

Type: Integer
Data: Percent Torque

8.17. DrvPctTorque

Type: Integer
Data: Drivers Percent Torque

8.18. TorqueMode

Type: TorqueModes
Data: Torque Mode

8.19. FuelRate

Type: Float
Data: Fuel Rate (liters / hour)

8.20. FuelUsed

Type: Float
Data: Total Fuel Used (liters)

8.21. HiResFuelUsed

Type: Float
Data: High Resolution Total Fuel Used (liters)

8.22. *IdleFuelUsed*

Type: Float
Data: Total Idle Fuel Used (liters)

8.23. *AvgFuelEcon*

Type: Float
Data: Average Fuel Economy (kilometers / liter)

8.24. *InstFuelEcon*

Type: Float
Data: Instant Fuel Economy (kilometers / liter)

8.25. *PrimaryFuelLevel*

Type: Integer
Data: Fuel Level (percent)

8.26. *SecondaryFuelLevel*

Type: Integer
Data: Fuel Level (percent)

8.27. *BrakeAppPressure*

Type: Float
Data: Brake Application Pressure (kPa)

8.28. *Brake1AirPressure*

Type: Float
Data: Brake Primary Air Pressure (kPa)

8.29. *Brake2AirPressure*

Type: Float
Data: Brake Secondary Air Pressure (kPa)

8.30. *OilTemp*

Type: Float
Data: Oil Temperature (Celsius)

8.31. *OilPressure*

Type: Integer
Data: Oil Pressure (kPa)

8.32. *IntakeTemp*

Type: Float
Data: Intake Temperature (Celsius)

8.33. *IntakePressure*

Type: Float
Data: Intake (Boost) Pressure (kPa)

8.34. TransTemp

Type: Float
Data: Transmission Temperature (Celsius)

8.35. CoolantTemp

Type: Float
Data: Coolant Temperature (Celsius)

8.36. CoolantLevel

Type: Float
Data: Coolant Level (0-100%)

8.37. CoolantPressure

Type: Float
Data: Coolant Pressure (kPa)

8.38. CurrentGear

Type: Float
Data: Transmission's Current Gear

8.39. SelectedGear

Type: Float
Data: Transmission's Selected Gear

8.40. BatteryPotential

Type: Float
Data: Battery Potential (Voltage)

8.41. VIN

Type: String
Data: Vehicle Identification Number

8.42. Make

Type: String
Data: Engine Make

8.43. Model

Type: String
Data: Engine Model

8.44. SerialNo

Type: String
Data: Engine Serial Number

8.45. UnitNo

Type: String
Data: Engine Unit Number

8.46. FaultCount

Type: Integer

Data: Number of Active Faults

8.47. CruiseSetSpeed

Type: Integer

Data: Cruise Control Set Speed (kph)

8.48. CruiseSwitches

Type: SwitchStates

Data: Cruise Control Switches

CruiseOnOff
CruiseSet
CruiseCoast
CruiseResume
CruiseAccel
CruiseActive

8.49. CruiseState

Type: CruiseControlStates

Data: Cruise Control State

8.50. GetFaultSPN

Type: Integer

Data: Fault SPN (or J1587 Sid/Pid)

8.51. GetFaultFMI

Type: Integer

Data: Fault FMI

8.52. GetFaultOccurrence

Type: Integer

Data: Fault Occurrence

8.53. GetFaultConversion

Type: Boolean

Data: Fault Conversion

8.54. IsJ1587Fault

Type: Boolean

Data: True – J1587 Fault
False – J1939 Fault

9. VEHICLE ENUMS

9.1. SleepModes

NoSleep
NA
WakeOnApp

9.2. ConnectionStates

NA
Initializing
Initialized
Discovering
Connecting
Connected
IsReady
Disconnecting
Disconnected
NotConnected
Reconnecting
Reconnected
NotReconnected
NotAuthenticated
J1708Restarting
ELDConnected
Heartbeat
DataChanged
CANFilterFull
DataTimeout
BluetoothTimeout
AdapterTimeout
AdapterRebooting
DataError
SystemError
Notification
AdapterMessage

9.3. SwitchStates

Off
On
Error
NA

9.4. CruiseControlStates

Off
Hold,
Accelerate,
Decelerate,
Resume,
Set,
AccelOverride,
NA

9.5. TorqueModes

LowIdleGovernor,
AccelPedal,
CruiseControl,
PTOGovernor,

RoadSpeedGovernor,
ASRControl,
TransControl,
ABSControl,
TorqueLimiting,
HighSpeedGovernor,
BrakingSystem,
RemoteAccelerator,
ServiceProcedure,
NotDefined,
Other,
NA

10. ELD Methods and Properties

10.1. ELD

Type: Class

Description: This is the ELD class that is instantiated by the API. All methods and properties are based in this class.

10.2. Connect

Parameters: None

Description: Connects the App to the Adapter for ELD recording. This must be called after the Adapter is connected.

10.3. IsCompatible

Type: Boolean
Data: Indicates whether the adapter is compatible with ELD recording.

Description: ELD recording requires Adapter firmware 3.10+. If the Adapter firmware is not at this level this value will be set to false and ELD recording will not be allowed.

10.4. DriverId

Type: String (0-22 characters)
Data: The driver id that will be recorded.

Description: The driver id is optional and if set will be included as the first ELD record when ELD recording is started. Note, the driver id is not persistent in the Adapter so it is the App's responsibility to save it.

10.5. ELDInterval

Type: Integer
Data: The interval for ELD recording (in minutes).

Description: Set this to record ELD records at a specific interval. The default is 60 minutes.

10.6. AlignELD

Type: Boolean
Data: Indicates whether ELD records should be aligned to the hour.

Description: Aligning the ELD records will ensure that an ELD record will be recorded at the top of the hour. An example of this would be an ELD interval of 15 minutes would record at the 15, 30, 45 minute mark and at the top of the hour. An interval that greater than an hour or one that cannot be aligned to the hour is not allowed if this option is selected.

10.7. RecordIFTA

Type: Boolean
Data: Indicates whether IFTA records should be recorded by the Adapter.

Description: Set this to have the Adapter record IFTA records along with ELD records. Note that recording IFTA records will reduce the available ELD recording time (ie. the number of records available for ELD recording).

10.8. IFTAInterval

Type: Integer
Data: The interval for IFTA recording (in minutes).

Description: Set this to record IFTA records at a specific interval. The default is 1 minute.

10.9. AlignIFTA

Type: Boolean
Data: Indicates whether IFTA records should be aligned to the hour.

Description: See AlignELD Description.

10.10. RecordStats

Type: Boolean
Data: Indicates whether Statistical records should be recorded by the Adapter.

Description: Set this to have the Adapter record Statistical records along with ELD records. Note that recording Statistical records will reduce the available ELD recording time (ie. the number of records available for ELD recording).

10.11. StatsInterval

Type: Integer
Data: The interval for Statistical recording (in minutes).

Description: Set this to record Statistical records at a specific interval. The default is 60 minutes.

10.12. AlignStats

Type: Boolean
Data: Indicates whether Statistical records should be aligned to the hour.

Description: See AlignELD Description.

10.13. IsHourAligned

Parameters:

Interval (Float)

Type: Boolean

Data: Returns whether the interval can be aligned to the hour.

Description: This is a helpful method to ensure that the intervals and aligns are correctly set.

10.14. IsStreaming

Type: Boolean

Data: Indicates whether the Adapter is to stream the ELD records to the App.

Description: This is set by the SetStreaming method.

10.15. SetStreaming

Parameters:

IsStreaming (Boolean)

Description: Instructs the Adapter to stream ELD records to the App.

10.16. IsSecured

Type: Boolean

Data: Indicates whether the ELD recording is secured to the App and Adapter.

Description: This is set by the SetSecured method.

10.17. SetSecured

Parameters:

IsSecured (Boolean)

Description: Instructs the Adapter to secure the ELD recording and not allow any other App to stop recording or upload or delete records.

10.18. IsAccessSecured

Type: Boolean

Data: Indicates whether ELD recording has been secured by another App.

Description: If this is set then the ELD recording has been started by another App and that App has secured the recording. You will only be allowed to stream the ELD records.

10.19. RecordingMode

Type: RecordingModes

Data: The RecordingMode for recording ELD data.

Description: This is the RecordingMode (see Enums) that determines if and how the ELD records will be recorded by the Adapter.

10.20. SetRecordingMode

Parameters:

RecordingMode (RecordingModes)

Description: Sets the RecordingMode used by the Adapter for recording ELD data.

10.21. SetRecordingMode

Parameters:

RecordConnected (Boolean)
RecordDisconnected (Boolean)

Description: Sets the RecordingMode based on the RecordConnected and RecordDisconnected parameters. This is a helper method for setting the RecordingMode.

10.22. IsRecordingLocally

Type: Boolean
Data: Indicates that the RecordingMode is set to record ELD records locally (App only).

Description: This is a helper property that references RecordingMode to indicate that the ELD records are only to be recorded by the App. The Adapter will not be doing any ELD recording. This is the same as IsStreaming true and RecordingMode set to RecordNever or RecordDisconnected.

10.23. IsRecordingConnected

Type: Boolean
Data: Indicates that the RecordingMode is set to record ELD only when the App is connected to the Adapter.

Description: This is a helper property that references RecordingMode to indicate that the ELD records are to be recorded by the Adapter only when it is connected to the App. This is the same as RecordingMode set to Always or RecordConnected.

10.24. IsRecordingDisconnected

Type: Boolean
Data: Indicates that the RecordingMode is set to record ELD only when the App is disconnected from the Adapter.

Description: This is a helper property that references RecordingMode to indicate that the ELD records are to be recorded by the Adapter only when it is disconnected from the App. This is the same as RecordingMode set to Always or RecordDisconnected.

10.25. IsStarted

Type: Boolean
Data: Indicates whether the Adapter has started ELD recording.

Description: This is set by the StartRecording and StopRecording methods. This will also indicated if the Adapter is recording at the time the App connects to it which will occur if the App disconnects without stopping ELD recording.

10.26. LocalRecordNo

Type: Integer
Data: The current local ELD record number.

Description: This is the record number of the current (ie. last) ELD record that was retrieved when streaming or recording locally. This will be updated whenever an ELD record is retrieved with IsStreaming true and RecordingMode set to RecordNever or RecordDisconnected (see property IsRecordingLocally).

10.27. CurrentRecordNo

Type: Integer
Data: The current Adapter ELD record number.

Description: This is the record number of the current (ie. last) ELD record that was recorded by the Adapter. This will be updated whenever an ELD record is retrieved with IsStreaming true and RecordingMode set to RecordConnected or RecordAlways.

10.28. RemainingTime

Type: Float
Data: The remaining time (in minutes) that is available for ELD recording.

Description: This is the duration that is remaining for all ELD recordings including IFTA and Statistical records. This is equivalent to the amount of ELD memory that is remaining. Note, that when there is no time remaining, ELD recording will stop. It is the responsibility of the application to monitor this and perform a Reset when necessary (see the Reset method). This will be updated whenever an ELD record is retrieved (streaming or direct request).

10.29. RemainingPercent

Type: Float
Data: The percentage of memory that is available for ELD recording.

Description: This is the percentage of ELD memory that is remaining for ELD recording (including IFTA and Statistical records). When the percentage is 10% or less, the adapter will pulsate a magenta color. When the percentage reaches zero the adapter will go solid magenta and stop recording. This will be updated whenever an ELD record is retrieved (streaming or direct request).

10.30. IsDataRetrieved

Type: Boolean
Data: Indicates that ELD records have been retrieved and ready for processing.

Description: Use this to process ELD records after retrieving them by streaming or direct record request.

10.31. GetRecord

Parameters:
Record Number (Integer)

Description: Retrieves the requested ELD record. The data will be returned in the ELD data property and an ELD IsDataRetrieved event will be raised.

10.32. DeleteRecords

Parameters:
Number of Records (Integer)

Description: Deletes the specified number of records starting at record number 1. Note that this does not erase the record from memory but just moves the first ELD record to the record after those deleted. Note also that because of the sequential nature of the ELD recordings, it is not possible to delete records in the middle of the recordings.

10.33. Reset

Parameters: None

Description: Deletes all ELD records and erases memory. This method should only be called when there is not enough ELD memory to store the required ELD records within the time remaining (see the RemainingTime and RemainingPercent properties). Caution, calling this method too often can affect the life of the ELD memory.

10.34. StartRecording

Parameters: None

Description: Instructs the Adapter to start ELD recording. Note, all recording parameters must be set prior to starting ELD recording.

10.35. StopRecording

Parameters: None

Description: Instructs the Adapter to stop ELD recording.

10.36. StartUpload

Parameters: None

Description: Instructs the Adapter to setup for retrieving ELD records.

Note: This must be called prior to retrieving ELD records (GetRecord method). If recording only while disconnected (IsRecordingDisconnected property) the Adapter will record records while the uploading is taking place and these records can be retrieved.

10.37. StopUpload

Parameters: None

Description: Instructs the Adapter to reset ELD record retrieval.

Note: This must be called after all ELD records have been retrieved. If recording only while disconnected (IsRecordingDisconnected property) the Adapter will stop recording records.

10.38. StartStreaming

Parameters: None

Description: Instructs the Adapter to start streaming the ELD records as they are being recorded. The data will be returned in the ELD data property and an ELD IsDataRetrieved event will be raised for each record received.

10.39. StopStreaming

Parameters: None

Description: Instructs the Adapter to stop streaming the ELD records.

11. ELD DATA

11.1. Record Number

Type: Integer
Data: The ELD record number (relative to 1)
Record Types: All

11.2. Record Id

Type: RecordIds
Data: Record Id (see RecordIds)
Record Types: All

11.3. Date

Type: DateTime
Data: The DateTime stamp of the record. When the App starts ELD recording, the current date-time is sent to the Adapter. This will persist in the Adapter for as long as the Adapter is powered. If the Adapter is power cycled, the current date-time will be set to 1/1/2042 UTC.
Record Types: All

11.4. LocalDate

Type: DateTime
Data: The local DateTime stamp of the record.
Record Types: All

11.5. Driver Id

Type: String
Data: Driver Identification – from the App
Record Types: ID

11.6. VIN

Type: String
Data: Engine VIN
Record Types: ID

11.7. Distance

Type: Float
Data: Vehicle Distance (miles) – from the Engine
Record Types: ELD, IFTA, Statistics

11.8. Odometer

Type: Float
Data: Vehicle Distance (miles) – from the OEM ECM
Record Types: ELD, IFTA

11.9. TotalHours

Type: Float
Data: Total Engine Hours
Record Types: ELD, Statistics

11.10. IdleHours

Type: Float
Data: Total Engine Idle Hours
Record Types: Statistics

11.11. TotalFuel

Type: Float
Data: Total Fuel Used (gallons)
Record Types: IFTA, Statistics

11.12. IdleFuel

Type: Float
Data: Total Idle Fuel Used (gallons)
Record Types: Statistics

11.13. Latitude

Type: Double
Data: GPS Latitude – This is only populated if GPS coordinates are available on the CAN bus. Note, some OEMs have on-board GPS and some off-board manufacturers such as Qualcomm may also publish GPS coordinates.
Record Types: ELD, IFTA

11.14. Longitude

Type: Double
Data: GPS Longitude – See Latitude.
Record Types: ELD, IFTA

11.15. Custom

Type: Byte Array
Data: 22 bytes.
Record Types: Custom

12. ELD ENUMS

12.1. RecordingModes

RecordNever (default)

ELD data will not be recorded by the Adapter (same as when RecordConnected and RecordDisconnected are both cleared.

RecordConnected

ELD data will be recorded by the Adapter only when the App is connected to the Adapter.

RecordDisconnected

ELD data will be recorded by the Adapter only when the App is disconnected from the Adapter.

RecordAlways

ELD data will always be recorded by the Adapter (same as when RecordConnected and RecordDisconnected are both set.

12.2. RecordId (Record Type)

NA (internal)

Used internally by the API.

DriverId (ID)

Received from the App before ELD is started.

VIN (ID)

Retrieved by the adapter when ELD is started.

StartEngine (ELD)

RPM > 0.

StartDriving (ELD)

Engine has started and speed >= 5 mph.

Driving (ELD)

Started driving or is driving and ELD interval has elapsed.

StopDriving (ELD)

Was driving and speed = 0 for at least 3 seconds.

StopEngine (ELD)

Stopped driving and RPM = 0.

IFTA (IFTA)

Started driving or is driving and IFTA interval has elapsed.

Stats (Statistics)

Started engine or is driving and Statistics interval has elapsed.

Custom (custom)

App defined record id. Custom record ids start at 128 (ie. custom record id 1 will be recorded as ELD record id 129.

13. Adapter LEDs

13.1. Normal Operation

Blue Blinking

The Adapter is waiting for the App to connect.

Blue Solid

The App is connected and the Adapter is receiving ECM data.

Green Blinking

The App is connected but the Adapter is waiting for the ECMs. Most likely this is because the key is off.

Red Blinking

The Adapter is in a user reset mode. This will occur one minute after plugging it in and not connecting to it. Unplug the adapter while it is blinking red, plug it back in, wait for it to blink red again, unplug it again, plug it back in and the adapter will reset itself to factory settings.

Red Solid

The Adapter is writing data to flash memory. DO NOT unplug it or you risk causing a factory reset.

13.2. ELD Operation

This only applies when the Adapter is recording data (RecordingMode set to RecordAlways, RecordConnected, or RecordDisconnected). If recording locally, the LEDs will function normally.

Blue Solid with Magenta Flashes

The App is connected and the Adapter is recording ELD data. The LEDs will flash Magenta each time a record is recorded.

Green Solid with Magenta Flashes

The App is not connected and the Adapter is recording ELD data. The LEDs will flash Magenta each time a record is recorded.

Green Pulsating

The App is not connected and the Adapter is recording ELD data but waiting for ECM data. Most likely the key is off.

Magenta Pulsating with Blue Flashes

The App is not connected and the Adapter is recording ELD data but the ELD date-time is not set. This is caused by power cycling the adapter after ELD recording has been started. It is recommended to connect the App to set the date-time. If the LEDs have been dimmed they will be brought to full brightness.

Magenta Pulsating with Cyan Flashes

This indicates that the ELD memory is almost full (90%). If the LEDs have been dimmed they will be brought to full brightness.

Magenta Solid

The ELD memory is full and ELD recording has stopped. If the LEDs have been dimmed they will be brought to full brightness.

14. Appendix

14.1. Common J1939 Sources

0 = Engine
3 = Transmission
11 = Brakes
23 = Instruments
25 = Climate Control
33 = Body
49 = Cab
85 = AfterTreatment
255 = Global

14.2. Common J1939 PGNs

57344 = Cab Message (1 s)
61443 = Engine Controller 2 (50 ms)
61444 = Engine Controller 1 (50 ms)
64777 = High Resolution Fuel Consumption (1 s)
64920 = AfterTreatment Information (On Request)
65110 = DEF Tank (1 s)
65198 = Air Pressure (1 s)
65203 = Fuel Information (On Request)
65213 = Fan Drive (1s)
65217 = High Resolution Vehicle Distance (1 s)
65226 = DM1 (Faults, 1 s)
65227 = DM2 (InActive Faults, On Request)
65228 = DM3 (InActive Fault Reset)
65229 = DM4 (Freeze Frames, On Request)
65235 = DM11 (Active Fault Reset)
65242 = Software Id (On Request)
65244 = Idle Operation (On Request)
65248 = Vehicle Distance (100 ms)

65253 = Engine Hours Revolutions (On Request)
65257 = Fuel Consumption (On Request)
65259 = Component Id (On Request)
65260 = Vehicle Id (On Request)
65262 = Engine Temperature (1 s)
65261 = Cruise Control Speed Setup (On Request)
65263 = Engine Fluid Level Pressure (500 ms)
65265 = Cruise Control Vehicle Speed (100 ms)
65266 = Fuel Economy (100 ms)
65269 = Ambient Conditions (1 s)
65270 = Inlet Exhaust Condition (500 ms)
65271 = Vehicle Electrical Power (1 s)
65272 = Transmission Fluids (1 s)
65274 = Brakes (1 s)
65276 = Dash Display (1 s)

14.3. Common J1587 MIDs

0 = Global
128 = Engine
130 = Transmission
136 = Brakes
140 = Instruments
142 = Vehicle Management
146 = Climate Control
171 = Driver Information
249 = Body Controller

14.4. Common J1587 PIDs

41 = Cruise Switches
44 = Fault Lamps
51 = Throttle Position
68 = Percent Torque
70 = Parking Brake
74 = Max Speed
84 = Speed
85 = Cruise Control
86 = Cruise Set Speed
91 = Accelerator Pedal Position
92 = Percent Load
96 = Fuel Level
100 = Oil Pressure
102 = Intake Pressure
105 = Intake Temperature
108 = Barometric Pressure
110 = Coolant Temperature
111 = Coolant Level
116 = Brake Application Pressure
117 = Brake Primary Pressure
118 = Brake Secondary Pressure
128 = Component Request

168 = Charging Voltage
175 = Oil Temperature
177 = Transmission Temperature
183 = Fuel Rate
184 = Instant Fuel Economy
185 = Average Fuel Economy
190 = RPM
192 = Multi-Section Message
194 = Diagnostics (On Request)
195 = Clear Fault
196 = Clear Fault Response
234 = Software Id (On Request)
235 = Idle Hours (On Request)
236 = Idle Fuel Used (On Request)
237 = VIN (On Request)
243 = Component Id (On Request)
245 = Distance
247 = Total Hours (On Request)
250 = Fuel Used (On Request)

14.5. Adapter Manual Factory Reset

14.5.1. Plug in the Adapter.
14.5.2. Wait one minute until the Adapter's LEDs go solid red.
14.5.3. When the Adapter starts blinking red, unplug it.
14.5.4. Plug the adapter back in.
14.5.5. The Adapter will go solid red immediately,
14.5.6. When the Adapter starts blinking red, unplug it again.
14.5.7. Plug the Adapter back in again.
14.5.8. The Adapter will go solid red and perform a factory reset.
14.5.9. Wait for the Adapter to start blinking blue.
14.5.10. DO NOT unplug the Adapter when it is SOLID RED.

15. Version Changes

15.1. Version 12.0

15.1.1. Initial Publication

15.2. Version 14.0

15.2.1. Minor improvements to the Bluetooth Discovery process.

15.2.2. Changed SetMaxConnectRetrys default to 10 seconds.

15.2.3. Changed SetDiscoveryTimeOut default to 30 seconds.

15.3. Version 19.0

15.3.1. Supports Adapter Firmware 3.7+.

15.3.2. Supports BLE (Bluetooth Low Energy) Adapters.

15.3.3. Added new Connection States:

15.3.3.1. Initializing

15.3.3.2. Initialized

15.3.3.3. Discovering

15.3.4. Added MinInterval property.

15.3.5. Added PerformanceMode property.

15.3.6. Added Get/Set PerformanceMode methods.

15.3.7. Added UseBLE, UseBT21, IsUsingBLE, and IsUsingBT21 properties.

15.4. Version 20.0

15.4.1. Renamed the Adapter class to the BlueFire class to avoid confusion with the Android Adapter widget.

15.4.2. Added GetTransmissionGears method that will retrieve the current and selected gears from the transmission ECM if the data is available.

15.4.3. Added an Adapter BT2.1 and a BLE checkbox to the Demo App that will select the appropriate Adapter type. Leaving both unchecked will cause the API to auto select the Adapter type.

15.4.4. Create two GitHub repositories, one for Eclipse development and one for Android Studio development.

15.5. Version 20.1

15.5.1. Removed IsUsingBT21 and IsUsingBLE properties in lieu of using properties UseBT21 and UseBLE which if not set will be set automatically by the API.

15.5.2. The Demo App Disconnect button will be shown immediately after connecting to allow for disconnecting while the API is attempting to discover an Adapter.

15.6. Version 20.2

15.6.1. The API will only raise Connection State 'Reconnected' when the Adapter is reconnected. Connection State 'AdapterConnected' will only be raised upon initial connection.

15.6.2. The Demo App has been modified to reflect the above 'Reconnected' Connection State.

15.6.3. Added method ResetAdapter.

15.6.4. The Demo App will edit for invalid hex characters in Send PGN Data.

15.6.5. Fixed fatal exception in SendPGN when sending data.

15.7. Version 21.0

15.7.1. Added optional Source and MID parameters to the GetVehicleInfo method.

15.7.2. Added optional Source and MID parameters to the GetFaults method.

15.7.3. Removed property IsFaultDataChanged.

15.7.4. Added Vehicle property IsJ1587Fault.

15.7.5. All methods take Source, PGN, and MID as integers.

15.7.6. Demo app shows Fault source.

15.8. Version 22.0

15.8.1. No longer compatible with Adapter Firmware less than 3.4 to 3. 6.

15.8.2. Renamed Version property to APIVersion.

15.8.3. Renamed IsVersionIncompatible to IsCompatible (logic change).

15.8.4. Removed GetSleepMode, GetLedBrightness, SetIgnoreDataBuses, GetIgnoreDatabuses and GetPerformanceMode as these are automatically called when the adapter connects.

15.8.5. All read-only properties have been changed to read-only method properties.

15.8.6. Get properties retrieve data from the adapter and Set properties send data to the adapter.

15.8.7. Added SetMaxReconnectAttempts method.

15.8.8. Added StopDataRetrieval method.

15.8.9. Renamed LastConnectedId to AdapterId.

15.8.10. Added Secure Adapter to SetSecurity and UpdateSecurity methods.

15.8.11. Added Connection States 'Authenticated' and 'NotAuthenticated'.

15.8.12. Adapter connection is now associated with connection state 'Authenticated' instead of 'AdapterConnected'.

15.8.13. Added ELD support for Adapter Firmware 3.10.x.

15.9. Version 22.1

15.9.1. Compatible with Adapter Beta Firmware 3.10.5.

15.9.2. Added SetTime method.

15.9.3. ELD records will only be written when appropriate.

15.9.4. Changed the LED display for ELD recording without a valid date to pulse magenta with blue recording blink.

15.9.5. Changed the LED display for ELD recording with almost full memory to pulse magenta with cyan recording blink.

15.10. Version 22.2

15.10.1. Compatible with API version 22.4.

15.10.2. Compatible with Adapter Beta Firmware 3.10.5.

15.10.3. Added GetEngineVIN method.

15.10.4. Added RetrievalMethod parameter to Vehicle Data methods.

15.11. Version 22.3

15.11.1. Compatible with API version 22.5.

15.11.2. Added property AndroidVersion.

15.11.3. Added property SetSyncTimeout.

15.11.4. Added ConnectionState CANFilterFull.

15.11.5. Renamed property DiscoveryTimeOut to DiscoveryTimeout.

15.11.6. Renamed property MaxConnectRetrys to MaxConnectAttempts.

15.11.7. Added method ELD.SetStreaming.

15.11.8. Added property ELD.LocalRecordNo.

15.11.9. Added ELD Enum RecordingModes.

15.11.10. Added property ELD.RecordingMode and method SetRecordingMode.

15.11.11. Added property ELD.IsRecordingLocally.

15.11.12. Added property ELD.IsRecordingConnected.

15.11.13. Added property ELD.IsRecordingDisconnected.

- *15.11.14.* Added to the Appendix instructions for manually resetting the Adapter.
- *15.11.15.* Included this document in the GitHub repositories.

15.12. Version 22.4

- *15.12.1.* Compatible with API version 22.6.
- *15.12.2.* Removed VIN from GetVehicleData method.
- *15.12.3.* Added Boolean return to Vehicle Data methods for synchronized calls.

15.13. Version 22.5

- *15.13.1.* The SetDiscoveryTimeOut method is renamed to SetDiscoveryTimeout (TimeOut to Timeout).
- *15.13.2.* Added SetAdvertisementTimeout method for use in very crowded BLE areas (like trade shows).
- *15.13.3.* For BLE adapters, if the ConnectToLastAdapter and SetSecurity (SecureAdapter) are not set, the API will connect to the adapter with the strongest signal.

15.14. Version 22.6

- *15.14.1.* Added HiRes and LoRes Distance and Odometer to Vehicle Data.
- *15.14.2.* Added GetDistance method which is the same as GetOdometer.
- *15.14.3.* GetDistance (and GetOdometer) will return the Engine Distance and the OEM Odometer (if available).
- *15.14.4.* Distance and Odometer are both in meters (Distance was in km).

15.15. Version 23.0

- *15.15.1.* Added property AdapterId.
- *15.15.2.* Added SetHeartbeatOn method
- *15.15.3.* Added SetNotificationsOn method.
- *15.15.4.* Added Notification ConnectionState.
- *15.15.5.* Added AdapterMessage ConnectionState.
- *15.15.6.* Added J1708Restarting ConnectionState.
- *15.15.7.* Added HardwareType property.
- *15.15.8.* Added OptimizeDataRetrieval property.
- *15.15.9.* Added SetOptimizeDataRetrieval method.
- *15.15.10.* Removed SetSyncTimeout method.
- *15.15.11.* Removed GetVIN and GetComponentId methods.
- *15.15.12.* Added GetVehicleVIN and GetVehicleId methods.
- *15.15.13.* Added StopRetrievingVehicleVIN and StopRetrievingVehicleId methods.
- *15.15.14.* Added ELD.StartUpload and StopUpload methods.
- *15.15.15.* Added ELD.LocalDate property.
- *15.15.16.* Renamed ELD.Time property to ELD.Date.
- *15.15.17.* Added transmission temperature (Vehicle.TransTemp) to method GetTemps.
- *15.15.18.* Added primary and secondary fuel gauge levels (Vehicle.PrimaryFuelLevel and SecondaryFuelLevel to GetFuelData.
- *15.15.19.* Better J1708 data retrieval with Firmware 3.13.
- *15.15.20.* No longer supports Android 4+ and Android 5+.
- *15.15.21.* Changed ELD method GetRecords to GetRecord.

15.16. Version 23.1

- *15.16.1.* Added property DeviceId.
- *15.16.2.* Added method SetDeviceId.

- *15.16.3.* Added DeviceId to the SetSecurity and UpdateSecurity methods.
- *15.16.4.* Removed the length limitation to Security UserName and Password.
- *15.16.5.* Updated the description in the SetSecurity and UpdateSecurity methods.
- *15.16.6.* Added a performance note to the UseBT21 property.

15.17. Version 23.2

- *15.17.1.* Added method SetBluetoothRecycleAttempt.
- *15.17.2.* Changed the MaxConnectAttempts default from 10 to 5.

15.18. Version 23.9

- *15.18.1.* Synchronized the documentation version with the API version.
- *15.18.2.* Added method SetIgnoreDatabuses.
- *15.18.3.* Added ConnectionState ELDConnected.
- *15.18.4.* Updated Authentication and other ConnectionStates.

15.19. Version 23.11

- *15.19.1.* Added property ConnectToLastAdapter.
- *15.19.2.* Added property MaxConnectAttempts.
- *15.19.3.* Added property MaxReconnectAttempts.
- *15.19.4.* Added property BluetoothRecycleAttempt.
- *15.19.5.* Added property BleDisconnectWaitTime.
- *15.19.6.* Added method SetBleDisconnectWaitTime.

15.20. Version 23.12

- *15.20.1.* Added property CANBusSpeed.
- *15.20.2.* Added ConnectionState J1939Starting.
- *15.20.3.* Added method GetKeyState.

15.21. Version 23.13

- *15.21.1.* Added property IgnoreJ1939.
- *15.21.2.* Added property IgnoreJ1708.
- *15.21.3.* Added method GetELDData.

15.22. Version 24.0

- *15.22.1.* Added property IsOBD2.
- *15.22.2.* Added property IgnoreOBD2.
- *15.22.3.* Added property SendAllPackets.
- *15.22.4.* Added property DisconnectedReboot.
- *15.22.5.* Added properties IsKeyOn and IsKeyOff.
- *15.22.6.* Renamed Connection State J1939Starting to CANStarting.
- *15.22.7.* Removed method UpdateSecurity.
- *15.22.8.* Removed method SetIgnoreDatabuses.

15.23. Version 24.1

- *15.23.1.* Added property Force500kBus.

15.24. Version 24.2

- *15.24.1.* Setting UseBLE and UseBT21 on will auto discover the Adapter.
- *15.24.2.* Property IgnoreJ1939 defaults to false.
- *15.24.3.* Property IgnoreJ1708 defaults to true.

15.24.4. Property IgnoreOBD2 defaults to true.

15.25. Version 24.3

15.25.1. Renamed the MonitorPGN method to StartMonitoringPGN.
15.25.2. Added IsBAMRTS parameter to the StartMonitoringPGN, StopMonitoringPGN and RequestPGN methods.
15.25.3. Changed SendPGN method parameters to (Source, PGN, Priority, PGN Data).

15.26. Version 25.0

15.26.1. Added ConnectionState AdapterReboot.
15.26.2. Removed ConnectionState AdapterConnected.
15.26.3. Renamed ConnectionState Authenticated to Connected.
15.26.4. Renamed ConnectionState CommTimeout to DataTimeout.
15.26.5. Renamed ConnectionState ConnectTimeout to BluetoothTimeout.
15.26.6. Added property ConnectAttempt.
15.26.7. Added property ReconnectAttempt.
15.26.8. Renamed property Message to ConnectionMessage.
15.26.9. Removed property ReconnectReason.
15.26.10. Removed method ClearMessages.

15.27. Version 25.1

15.27.1. Added method KeepJ1708PortOpen.

15.28. Version 25.2

15.28.1. Added Interval to SendPGN.

15.29. Version 26.0

15.29.1. Removed the Eclipse GitHub link.
15.29.2. Changed "Truck" to "Vehicle".
15.29.3. Added ConnectionState IsReady.
15.29.4. Updated the ConnectionState Connected description.
15.29.5. Updated the ResetAdapter method description.
15.29.6. Updated the IsCANAvailable and IsJ1708Available properties.