Tritium OCPP 1.6J

Client Documentation

Version 4

10 January 2018

Brisbane, Australia
http://www.tritium.com.au

Contents

1 Abbreviations

Abbreviation	Description
AC	Alternating Current
CCS	Combined Charging System
CPO	Charge Point Operator
CSL	Comma Separated List
DC	Direct Current
EV	Electric Vehicle
EVSE	Electric Vehicle Supply Equipment
HMI	Human Machine Interface
OCPP	Open Charge Point Protocol
R	Read Only
RW	Read/Write

2 Initial Setup

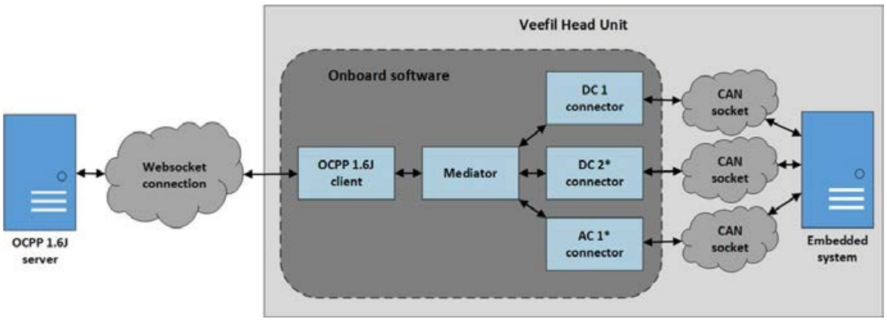
Customers should communicate to Tritium their preferred websockets URL prior to shipping so that their charger arrives and connects immediately to the correct server once it is powered on. If at any time the customer wishes to change the websockets URL after installation they can do so using the *changeConfigurationRequest* packet that is part of the OCPP 1.6 protocol, in particular they should set the *CentralURL* configuration parameter.

Passwords and security recommendations are outlined in another document – please refer to our technical white paper, *Staying Ahead: OCPP 1.6, OCPP 1.6 Deployment Procedures for Tritium Veefil Charge Points*.

3 Architectural Summary

3.1 Single Charger Systems

The following figure depicts the architecture currently deployed in single unit systems. Note that the AC backpack is an optional add-on to the standard unit, and sometimes only a single DC connector will be installed. Also note that if more than one DC charging cable is installed, then they will be mutually exclusive in that only one of them can be in operation at any given time. For more information on mutually exclusive connectors, see section *6 Mutually Exclusive Connectors*.

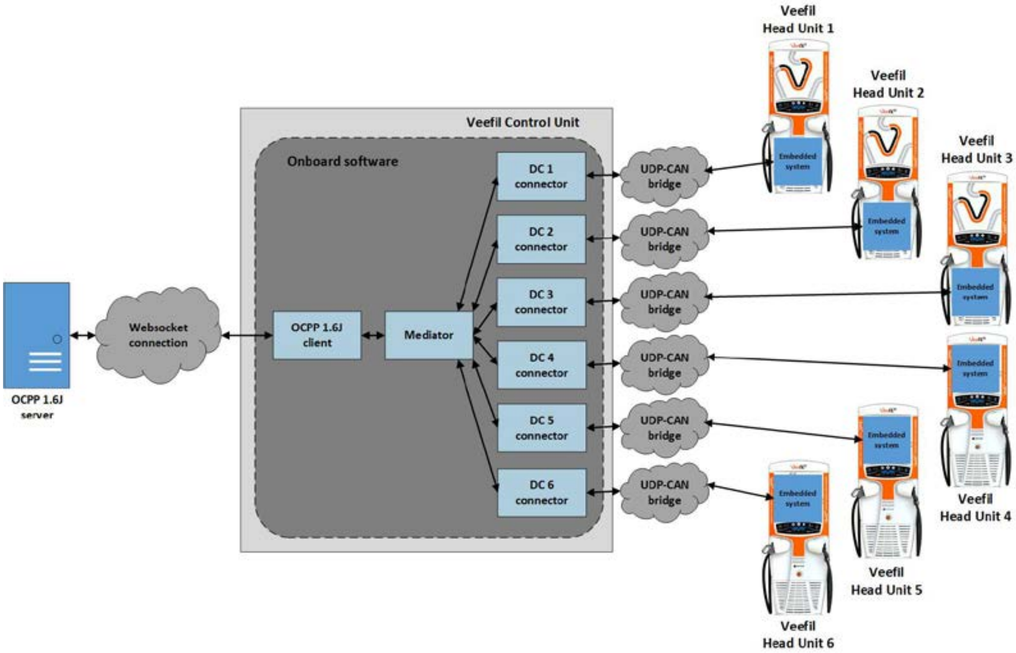


*Connectors marked with an asterisk may be optional or not present on the unit.

The number of unique connectors on a single charging unit can be more than one. In any case, there will be a single OCPP connection to the back end. Each charging connector will be uniquely and reliably numbered.

3.2 Grouped Charger Systems

The following figure summarises the interaction of the various components of the group site as they pertain to the OCPP protocol and how information is coordinated with the head units. As in the case of the single charger system, there will be a single OCPP connection to the back end. Again, each charging connector will be uniquely and reliably numbered.



4 Measurands

The EVSE sends the *MeterValuesRequest* and the *StopTransactionRequest* packets to the CPO in varying circumstances, and these packets convey "measurands", or telemetry related information.

4.1 Supported Measurands

The following tables lists the measurands supported in the Tritium implementation of the OCPP 1.6 specification.

Measurand	Specification Description	Comments
Current.Import	Instantaneous current flow to EV	Reported in Amperes
Current.Offered	Maximum current offered to EV	Reported in Amperes.
Energy.Active.Import.Register	Energy imported by EV (Wh or KWh)	Reported in Watt-hours.
Energy.Active.Import.Interval	Energy imported by EV	Reported in seconds
Frequency	Instantaneous reading of powerline frequency	Reported in hertz.
Power.Active.Import	Instantaneous active power imported by EV. (W or kW)	Reported in Watts.
Power.Offered	Maximum power offered to EV	Reported in Watts.
RPM	Pump speed in RPM	Average speed of relevant coolant pumps.
SoC	State of charge of charging vehicle in percentage	Reporting in range [0.0, 100.0].
Temperature	Temperature	Temperature of the coolant as it returns from the radiator.
Voltage	Instantaneous AC RMS supply voltage	Reported in Volts.

4.2 Measurand Sources

The OCPP specification describes four different pathways in which the CPO can receive measurand conveying updates. They are part of the Core Profile in the standard configuration, and briefly described as follows:

1. *MeterValuesSampleData* - to be sent periodically during a charging session based the *MeterValueSampleInterval* parameter.
2. *MeterValuesAlignedData* - to be sent periodically based on the *ClockAlignedDataInterval* parameter. To be associated with a distinct connector.

However, accumulated and averaged across all charging sessions over the clock alignment interval.

3. *StopTxnSampledData* - to be sent at the end of a charging session, but collected at even intervals during the charging session according to the *MeterValueSampleInterval* parameter.
4. *StopTxnAlignedData* - to be sent at the end of a charging session, but collected at even intervals during the charging session according to the *ClockAlignedDataInterval* parameter.

In most cases a packet sent from the EVSE to the CPO will convey measurands that pertain to a specific charging session on a distinct connector. The exception is in the case of *MeterValuesAlignedData*, where measurand values can be averaged and/or accumulated across charging sessions. Note that important configuration keys related to transmission of measurands are the *ClockAlignedDataInterval* and *MeterValueSampleInterval* (see the configuration section) parameter, both of which describe time intervals.

Each of the four measurand sources as listed above can be configured to convey a custom selection of measurand values (see the configuration section). These four different data sources fall in to two categories, sampled and aligned data.

4.3 Sampled Data

For the most part, sampled data represents the current state of the measurand at the time of the sampling operation, it represents that state of something at a precise point in time. For example, the value reported for the *Energy.Active.Import.Register* measurand is always the total accrued energy imported into the EV, as part of a charging session, at a given point in time.

4.4 Aligned Data

For almost all supported measurands, such as temperature or RPM, the value reported as aligned data is the (time weighted) average value throughout the clock aligned interval. However, for other measurands the value reported will equate to the total accumulated during the given period (starting from zero). For example, the value reported for the *Energy.Active.Import.Register* measurand will equate to the total energy transferred during the interval. Similarly for the *Energy.Active.Import.Interval* measurand, the value reported it is the total amount of time the EVSE spent charging during the aligned data interval. All other supported measurands are averaged. Customers can control the smoothness of the averaging operation by configurating a non-zero value for the *ClockAlignedAccrualInterval* parameter, which is described in the Support Profile of the configuration section.

5 Status Notification

The EVSE sends *StatusNotificationRequests* to the CPO whenever the state of a connector changes. The CPO can at any time force the EVSE to send a *StatusNotificationRequest* using the *TriggerMessageRequest* packet.

5.1 Charge Point Status

The EVSE will send *StatusNotificationRequests* to CPO using an appropriate charge point status as listed in the following table.

Status	Tritium Specific Conditions
Available	The connector is functional. The plug is holstered.
Preparing	The connector is unholstered. The connector is plugged in to the EV prior to a charging session. Authorization may be taking place.
Charging	The EVSE is engaged in the operation of charging the EV.
Finishing	The charging session has stopped but the connector has not yet been unlocked, or has not yet been returned to the holster.
Reserved	The connector is reserved.
Unavailable	The connector has been made unavailable the CPO back end.
Faulted	The EVSE has experienced an error condition and the connector cannot be used to charge an EV.

This is a subset of the complete list of statuses that may be reported (we do not report *SuspendedEVSE* or *SuspendedEV* in the current release). The EVSE will send status updates only when state changes accordingly, or when forced to by the receipt of an appropriate *TriggerMessageRequest*, or upon the (re)establishment of connectivity to the CPO.

5.2 Connector 0

As per the specification, connector 0 can only send *Available*, *Unavailable* or *Faulted*. In the case of there being more than one connector in the EVSE, the Tritium implementation will always return the best-case scenario. That is, if connector 1 is *Faulted* and connector 2 is *Available*, connector 0 will be reported as available.

6 Mutually Exclusive Connectors

The standard Veefil-RT unit ships with support for both the CHAdeMO and CCS DC protocols, and this presents as two physically distinct connectors on the one EVSE. Note however, that the operation of these two connectors is mutually exclusive. If one is in operation the other is unavailable.

6.1 Status Notification

A changing status of one of a set of mutually exclusive connectors has the potential to cause the status of the other connector to change. This relationship is defined in the following table (where Connector 1 and Connector 2 are mutually exclusive connectors):

Newly established status of Connector 1	Status reported by Connector 2 given that it would otherwise be available
Available	Available
Preparing	Unavailable
Charging	Unavailable
Finishing	Unavailable
Reserved	Reserved
Unavailable	Unavailable
Faulted	Faulted

Note that it is only the changes in status which are conveyed from the EVSE to the CPO

6.2 Reservation

Note that reserving any one of a set of mutually exclusive connectors equates to reserving them all.

6.3 Change Availability

Note that changing the availability of any one of a set of mutually exclusive connector equates to changing the availability of all of them.

7 Diagnostics

Occasionally Tritium may request that the customer generate diagnostic data for problem solving purposes. This is done with the *getDiagnosticsRequest* packet. Note however that the diagnostic information generated by this operation is encrypted and can be unencrypted with the assistance of Tritium staff.

8 Firmware Updates

Tritium may occasionally alert customers to the availability of firmware updates that contain improvements and/or bug fixes. In order to install these firmware updates the client issues the *updateFirmwareRequest* packet to the onboard OCPP client, which takes care of downloading and installing the updates. In such a case, Tritium will coordinate with the customer to ensure the setting of the *location* field (which describes the URI location of the download bundle) in the *updateFirmwareRequest* is done correctly. A firmware update will cause the system to reboot.

9 Error Codes

Error codes are distributed using the *vendorErorrCode* field of the *StatusNotificationRequest* packet. When an error state is encountered, the connector will report a *Faulted* status. Customers should expect the *vendorErrorCode* to present as a comma separated list of up to four integers values, embodied as a string. Some examples would be

- "132,23,0,0"
- "63,0,0,0"
- "63,37,138,45"

See https://helium.office.tritium.com.au/vfv/Documents/ErrorCodes.htm for a complete description of error codes.

10 Smart Charging

10.1 Grouped Chargers

For grouped chargers, Tritium has developed an optimal power distribution system based on fairness and efficiency. Presently, we only support the setting of the maximum power consumption of the site in its entirety. Therefore, it is only *setChargingProfileRequest* messages pertaining to connector Id 0 that can influence the power management of the site. To get started, we recommend that clients pack the fields of the *setChargingProfileRequest* packets using the following table as a guide.

Parameter	**Value**
connectorId	0
csChargingProfiles.chargingProfilePurposeType	ChargePointMaxProfile
csChargingProfiles.chargingProfileKindType	Absolute
csChargingProfiles.chargingProfileId	1
csChargingProfiles.stackLevel	0
csChargingProfiles.chargingSchedule.startSchedule	UTC now
csChargingProfiles.chargingSchedule.chargingRateUnit	W
csChargingProfiles.chargingSchedule.chargingSchedulePeriod[0].startPeriod	0
csChargingProfiles.chargingSchedule.chargingSchedulePeriod[0].limit	1200000

Any field that is optional, and is not mentioned in this table, is implicitly recommended to be left unspecified. The effect of sending a *setChargingProfileRequest* according to the above table will impose a limit of 1.2MW on the entire site, until otherwise notified.

Note that Tritium supports recurring (all) profile kind types for those customers wanting to explore the more advanced features of the smart charging profile.

10.2 Connector Specific Load Balancing

Future revisions will include support for per connector load balancing in the group context.

11 Configuration

The CPO uses the *getConfigurationRequest* and *changeConfigurationRequest* commands to read and write configuration values.

There are four profiles described in the specification, and as supported by Tritium, namely the *Core*, *Local Auth List Management*, *Reservation*, and *Smart Charging* profiles. In addition to this, Tritium provides many custom configuration keys, these have been divided into the *Support*, *Veefil*, *CCU*, *CCUFlash*, and *ACUFlash* profiles. In this section, we define the extent of the support provided in the context of each profile. All configuration values are persisted.

11.1 Core Profile

The *Core Profile* is for the most part supported.

Key	**R/ RW**	**Type**	**Default**	**Specification description**	**Support (Y/N), Comments**
AllowOfflineTxForUnknownId	RW	Boolean	False	If this key exists, the Charge Point supports Unknown Offline Authorization. If this key reports a value of true, Unknown Offline Authorization is enabled.	*Y*
AuthorizationCacheEnabled	RW	Boolean	False	If this key exists, the Charge Point supports an Authorization Cache. If this key reports a value of true, the Authorization Cache is enabled.	*Y*
AuthorizeRemoteTxRequests	R or RW (RW)	Boolean	False	Whether a remote request to start a transaction in the form of a RemoteStartTransaction.req message should be authorized beforehand like a local action to start a transaction.	*Y* *Read-write.*
BlinkRepeat	RW	Integer	-	Number of times to blink Charge Point lighting when signalling.	*N* *Optional key.*
ClockAlignedDataInterval	RW	Integer (seconds)	900	Interval in seconds to transmit "clock aligned" data - for a description of clock aligned data see the original documentation	*Y*
ConnectionTimeOut	RW	Integer (seconds)	60	Interval (from successful authorization) until incipient charging session is automatically canceled due to failure of EV user to (correctly) insert the charging cable connector(s) into the appropriate connector(s).	*Y*
GetConfigurationMaxKeys	R	Integer	128	Maximum number of requested configuration keys in a GetConfiguration.req.	*Y*
HeartbeatInterval	RW	Integer	10	Interval of inactivity (no OCPP exchanges) with central system	*Y*

Key	R/ RW	Type	Default	Specification description	Support (Y/N), Comments
				after which the Charge Point should send a Heartbeat.req.	
LightIntensity	RW	Integer (%)	-	Percentage of maximum intensity at which to illuminate Charge Point lighting	*N* *Optional key. Light intensity is automated with sensors.*
LocalAuthorizeOffline	RW	Boolean	False	Whether the Charge Point, when offline, will start a transaction for locally-authorized identifiers.	*Y*
LocalPreAuthorize	RW	Boolean	False	Whether the Charge Point, when online, will start a transaction for locally-authorized identifiers without waiting for or requesting an Authorize.conf from the Central System.	*Y*
MaxEnergyOnInvalidId	RW	Integer	-	Maximum energy in Wh delivered when an identifier is invalidated by the Central System after start of a transaction.	*N* *Optional key.*
MeterValuesAlignedData	RW	CSL	"Temperature,RPM"	Clock-aligned measurand(s) to be included in a MeterValues.req PDU, every ClockAlignedDataInterval seconds	*Y*
MeterValuesAlignedDataMaxLength	RW	Integer	11	Maximum number of items in a MeterValuesAlignedData Configuration Key.	*Y*
MeterValuesSampledData	RW	CSL	"Power.Active.Import, Energy.Active.Import.Register,SoC" “Current.Offered” “Current.Import”	Sampled measurands to be included in a MeterValues.req PDU, every MeterValueSampleInterval seconds. Where applicable, the Measurand is combined with the optional phase; for instance: Voltage.L1 Default: "Energy.Active.Import.Register"	*Y*
MeterValuesSampledDataMaxLength	RW	Integer	11	Maximum number of items in a MeterValuesSampledData Configuration Key.	*Y*
MeterValueSampleInterval	RW	Integer (seconds)	10	Interval between sampling of metering (or other) data, intended to be transmitted by "MeterValues" PDUs. For charging session data (ConnectorId>0), samples are acquired and transmitted periodically at this interval from the start of the charging transaction. A value of "0" (numeric zero), by convention, is to be interpreted to mean that no sampled data should be transmitted.	*Y*

Key	R/ RW	Type	Default	Specification description	Support (Y/N), Comments
MinimumStatusDuration	RW	Integer	-	The minimum duration that a Charge Point or Connector status is stable before a StatusNotification.req PDU is sent to the Central System.	*N* *Optional key.*
NumberOfConnectors	R	Integer	Factory set	The number of physical charging connectors of this Charge Point.	*Y*
ResetRetries	RW	Integer	3	Number of times to retry an unsuccessful reset of the Charge Point.	*Y*
ConnectorPhaseRotation	R	CSL	NotApplicable	The phase rotation per connector in respect to the connector's energy meter (or if absent, the grid connection). Possible values per connector are: NotApplicable (for Single phase or DC Charge Points)	*Y*
ConnectorPhaseRotationMaxLength	R	Integer	-	Maximum number of items in a ConnectorPhaseRotation Configuration Key.	*N* *Optional key.*
StopTransactionOnEVSideDisconnect	R or RW (R)	Boolean	True	When set to true, the Charge Point SHALL administratively stop the transaction when the cable is unplugged from the EV.	*Y* *Read only.*
StopTransactionOnInvalidId	R or RW (R)	Boolean	True	whether the Charge Point will stop an ongoing transaction when it receives a non-Accepted authorization status in a StartTransaction.conf for this transaction.	*Y* *Read only.*
StopTxnAlignedData	RW	CSL	""	Clock-aligned periodic measurand(s) to be included in the TransactionData element of StopTransaction.req MeterValues.req PDU for every ClockAlignedDataInterval of the charging session	*Y*
StopTxnAlignedDataMaxLength	R	Integer	11	Maximum number of items in a StopTxnAlignedData Configuration Key.	*Y*
StopTxnSampledData	RW	CSL	""	Sampled measurands to be included in the TransactionData element of StopTransaction.req PDU, every MeterValueSampleInterval seconds from the start of the charging session	*Y*
StopTxnSampledDataMaxLength	R	Integer	11	Maximum number of items in a StopTxnSampledData Configuration Key.	*Y*

Key	R/ RW	Type	Default	Specification description	Support (Y/N), Comments
SupportedFeatureProfiles	R	CSL	"Core,FirmwareManagement,LocalAuthListManagement,Reservation,RemoteTrigger,SmartCharging"	A list of supported Feature Profiles. Possible profile identifiers: Core, FirmwareManagement, LocalAuthListManagement, Reservation, SmartCharging and RemoteTrigger.	*Y*
SupportedFeatureProfilesMaxLength	R	Integer	7	Maximum number of items in a SupportedFeatureProfiles Configuration Key.	*Y*
TransactionMessageAttempts	RW	Integer	1	How often the Charge Point should try to submit a transaction-related message when the Central System fails to process it.	*Y*
TransactionMessageRetryInterval	RW	Integer (seconds)	10	How long the Charge Point should wait before resubmitting a transaction- related message that the Central System failed to process.	*Y*
UnlockConnectorOnEVSideDisconnect	R or RW (R)	Boolean	True	When set to true, the Charge Point SHALL unlock the cable on Charge Point side when the cable is unplugged at the EV.	*Y* *Read only.*
WebSocketPingInterval	RW	Integer	60	Only relevant for websocket implementations. 0 disables client side websocket Ping/Pong. In this case there is either no ping/pong or the server initiates the ping and client responds with Pong. Positive values are interpreted as number of seconds between pings. Negative values are not allowed. ChangeConfiguration is expected to return a REJECTED result.	*Y*

11.2 Local Auth List Management Profile

This profile is supported.

Key	R/RW	Type	Default	Specification description	Support (Y/N), Comments
LocalAuthListEnabled	RW	Boolean	True	Whether the Local Authorization List is enabled.	Y
LocalAuthListMaxLength	R	Integer	16384	Maximum number of identifications that can be stored in the Local Authorization List	Y
SendLocalListMaxLength	R	Integer	1024	Maximum number of identifications that can be send in a single SendLocalList.req	Y

11.3 Reservation Profile

This profile is supported; however, we are yet to integrate these changes into the HMI. For more information please contact customer support.

Key	**R/RW**	**Type**	**Default**	**Specification description**	**Support (Y/N), Comments**
ReserveConnectorZero Supported	R	Boolean	False	If this configuration key is present and set to true: Charge Point supports reservations on connector 0.	Y

11.4 Smart Charging Profile

This profile is supported.

Key	**R/RW**	**Type**	**Default**	**Specification description**	**Support (Y/N), Comments**
ChargeProfileMaxStack Level	R	Integer	32	Max StackLevel of a ChargingProfile. The number defined also indicates the max allowed number of installed charging schedules per Charging Profile Purposes.	*Y*
ChargingSchedule AllowedChargingRateUnit	R	CSL	"Power"	A list of supported quantities for use in a ChargingSchedule. Allowed values: Current' and 'Power'	*Y*
ChargingScheduleMax Periods	R	Integer	64	Maximum number of periods that may be defined per ChargingSchedule.	*Y*
ConnectorSwitch3to1 Phase Supported	R	Boolean	False	If defined and true, this Charge Point support switching from 3 to 1 phase during a charging session.	*Y*

Key	**R/RW**	**Type**	**Default**	**Specification description**	**Support (Y/N), Comments**
MaxChargingProfiles Installed	R	Integer	512	Maximum number of Charging profiles installed at a time	*Y*

11.5 Support Profile

Note the support profile is added by Tritium specifically to compliment the OCPP standard configuration keys.

Key	R/RW	Type	Default	Specification description
AuthorizationKey	W	String		See the OCPP-J-1.6 specification, basic HTTP authorization.
Supported Measurands	R	CSL	"Current.Import,Current.Offered,Energy.Active.Import.Register,Energy.Active.Import.Interval,Frequency,Power.Active.Import,Power.Offered,RPM,SoC,Temperature,Voltage"	Obtain the list of supported “measurands” - this is the set of meter related values supported by the underlying implementation. This is so that the values that may be supplied for the MeterValuesAlignedData, MeterValuesSampledData, StopTxnSampledData, and StopTxnAligneddData keys in the Core Profile is explicit and knowable.
ClockAligned AccrualInterval	RW	Integer (seconds)	10	An internal timer used for accruing aligned data associated with the MeterValuesAlignedData and StopTxnAligneddData keys in the Core Profile. Increases the granularity of the sampling and provides a smoother average. A value of 0 disables this.
ReconnectInterval	RW	Integer (seconds)	10	The time in seconds to wait before reconnecting upon Websocket disconnection. See ReconnectExpBackOffMax.

Key	R/RW	Type	Default	Specification description
ReconnectExp BackOffMax	RW	Integer	4	The number of times the ReconnectInterval should double when the Websocket connection is consistently failing. A value of 10 for the ReconnectInterval and a value of 4 for the ReconnectExpBackOffMax would mean that the reconnection interval would be 10, 20, 40, 80, 160, 160, 160, but only if every connection was consistently failing. The reconnection interval is reset after every successful connection.
StopTxnEnd SampledData	RW	CSL	"Power.Active.Import, Energy.Active.Import. Register,SoC"	Sampled measurands to be sent in a with the Tx.End.
MeterValues AlignedOnlyZero	RW	Boolean	True	Whether MeterVauesAlignedData should only be sent for Connector Id 0.
CentralURL	RW	String	Factory set	Url of the provider backend.

11.6 Veefil Profile

This profile contains a set of keys that will appear to customers who are using Veefil units. Some customers using the Tritium OCPP 1.5 software bundle will be familiar with these configuration keys, but will be accustomed to accessing them using different naming conventions. See the comments for more details.

Key	R/RW	Type	Default	Specification description	Comments
Serial Number	R	String	Factory set	Serial Number of the Veefil.	Previously *serial*.
Software Version	R	String	Factory set	Version of software presently installed on the Veefil.	Previoiusly *software_version*.
AutoStart Remote Charge	RW	Boolean	False	True: On receipt of a RemoteStart Transaction, the Veefil will automatically start charging the EV as soon as it is able.	Previously *start_charge_ remote*.

Key	R/RW	Type	Default	Specification description	Comments
Lock Remote Charge	RW	Boolean	False	This field applies to circumstances relating to the receipt of a remoteStart Transactio Request. True: The HMI will become locked soon after the charge has started and may only be unlocked by a swipe of the correct RFID, as per the normal user interaction. False: The HMI does not lock. RECOMMENDED TO BE SET False, as loss of communication will result in the inability to unlock connector from the charger until car is full or the emergency stop mechanism is triggered.	Previously *lock_after_remote*.
FreeMode RFID	RW	String	"00000000"	Hexadecimal Card ID used whenever the charger is engaged in a free charging session. Padded to minimum of 8 or 14 characters (4 and 7 byte cards)	Previously *free_mode_rfid*.
PriceKWh	RW	Float	0.0	If displaying price/cost information, this is the orice cost per kilowatt hour in cents. 0: Feature disabled	Previously *price_kWh*.
PriceMinute	RW	Float	0.0	If displaying price/cost information, this is the price cost per minute in cents. 0: Feature disabled	Previously *price_minute*.

Key	R/RW	Type	Default	Specification description	Comments
PriceDisplay Interval	RW	Integer (seconds)	3	If displaying price/cost information, this is the duration that the front panel will pause to display information before switching back to the original display (max 99s)	Previously *price_display*.
ModemAPN	RW	String	Factory set	Modem network APN provider	Previously *modem_apn*.
ModemUser	RW	String	Factory set	Modem network Username	Previously *modem_user*.
Modem Password	RW	String	Factory set	Modem network Password	Previously *modem_pw*.
Modem Ccid	R	String	Factory set.	Modem CCID	*New*
Modem Imsi	R	String	Factory set	Modem IMSI.	*New*
Modem Settings Apply	W	Boolean		A non persistent flag/triggering mechanism for applying ModemAPN, ModemUser and ModemPassword to the cellular modem	
PingHost	RW	String	"google.com"	Location of a machine that can be pinged by the wireless connection, this can be an internal machine if using a secure VPN. Please note this is not a webpage thus a http address will be rejected	Previously *ping_host*.
Front Panel Update Millis Interval	RW	Integer (milliseconds)	1000	If displaying price/cost information, this is the regularity with which to update the cost displayed to the user in the front panel, in milliseconds	*New*
Model	R	String	Factory set	The unit model.	*New*
Vendor	R	String	"Tritium"	The vendor.	*New*

11.6.1

11.6.2 Displaying Price Information

The Veefil unit comes with a limited ability to display price information to the user. The component of the HMI that displays "Battery %" can be configured to oscillate between displaying the battery percentage, and the cost of the charge. The period of oscillation is governed by the *PriceSecondsInterval* parameter; however, it is enabled only if one of the *PriceMinute* or *PriceKWh* parameters is non zero. The equation used to calculate the price displayed is

PriceDisplayed = (PriceKWh)*(Energy imported by EV in KWh) + (PriceMinute)*(Time of charging session in minutes)

The price displayed will have zero, one, or two decimal places depending on the magnitude of the value to be displayed. This is due to the limitation of there being only three seven segment displays to show this number. Some examples of the price that will be displayed are given in the table below.

PriceKWh	Energy Imported (KWh)	PriceMinute	Time in Minutes	Displayed Value
10	5	1	3	0.53
10	10	1	6	1.06
10	100	1	60	10.6
10	1000	1	600	106
10	10000	1	6000	---[1]

1 – As soon as the value of the calculation is greater than or equal to 1000 the HMI will no longer display the cost of the transaction, instead it will display "---".

11.6.3 The Modem Parameters

The Modem parameters only apply if the unit comes with a configurable modem installed. In such a case, the client generally will set *ModemAPN*, *ModemPassword*, and *ModemUser*, and once satisfied these are correct, will send the configuration key *ModemSettingsApply* with the value *True.* This will cause the settings to be applied to the on-board modem.

ModemSettingsApply is a write only Boolean that is not stored.

Note that *ModemCcid* and *ModemImsi* are read only values which are set in factory.

11.7 Advanced Profiles

The advanced profiles listed here relate to changing configuration settings within the embedded system inside the charging unit.

11.7.1 CCU Profile

This profile contains a set of keys that will appear to customers who are using Veefil units. This profile pertains to an isolated embedded component in the Veefil unit, namely the CCU. A distinguishing characteristic of this profile is that no reboot is required in order to change a given value.

Key	R/RW	Type	Default	Specification description	Comments
HU1.CCURFIDDisable	RW	Boolean	False	True: RFID Disabled allowing free charge False: RFID Enabled, authentication required to charge.	Previously *rfid_disable*.
HU1.CCUDock **Connectors** **Disable**	RW	Boolean	False	True: Connectors do not need to be returned to the charger between charging sessions. False: Connectors need to be return to charger between charging sessions	Previously *dock_connectors_disable*.
HU1.CCUMaxHalf **Enable**	RW	Boolean	False	True: Max/Half button will be enabled with 95%/80%. CCUSOCLimit Disable must be set to false for this option to take effect False: Max/Half button disabled	Previously *max_half_enable*.
HU1.CCUSOCLimit **Disable**	RW	Boolean	True	True: Users can charge their cars to 100% False: Maximum charge is 80% (unless max_half_enable is set to true)	Previously *SOC_limit_disable*.

Key	R/RW	Type	Default	Specification description	Comments
HU1.CCUChargeTime CountUp	RW	Boolean	True	True: Front panel displays charge time elapsed False: Front panel displays time to charge finish (estimate)	Previously *charge_time_count_up*.
HU1.CCUHasACBack Pack	R	Boolean	Factory set	True if charger has AC charging option installed, false otherwise.	Previously *has_ac_backpack*.
HU1.CCUHolster Reversed	RW	Boolean	False	Flap sensors have been configured in reverse.	*New*

11.7.2 CCUFlash Profile

This profile contains a set keys that will appear to customers who are using Veefil units. This profile pertains to an isolated embedded component in the Veefil unit, namely the CCU. A reboot of the CCU is required to enforce a change in any of these values. Therefore, these values can only be changed when the Veefil unit is not charging.

Key	R/RW	Type	Default	Specification description	Comments
HU1.CCUFlashDC MaxOutputPower	RW	Integer	Factory set.	Max power charger outputs during charging. Not intended for dynamic grid power loading, will also enforce 5-minute time out when changed	Previously *dc_max_output_pwr*.
HU1.CCUFlashDC MaxChargeTime	RW	Integer	0	Max time in minutes per charge session. (max 200 minutes) 0: Feature disabled	Previously *dc_max_charge_time*.
HU1.CCUFlashDC MaxGridCurrent	RW	Integer	255	200-255 – Disabled Initial current will be limited by max_output_pwr, then ramp down quickly to max_grid_current. Not intended for dynamic grid power loading, will also enforce 5-minute time out when changed.	Previously *dc_max_grid_current*.
HU1.CCUFlashDC Breaker Derating	RW	Integer	255	255 – Disabled Setpoint between 0 – 100. Allows reduction of grid current setpoint to compensate for the	Previously *dc_breaker_derating*.

Key	R/RW	Type	Default	Specification description	Comments
				effect of internal charger temperature on the tripping point of the internal circuit breaker. Only required on chargers that are operating close to their breaker rating. Grid current will be at the "dc_max_grid_current" setpoint when at or below 25°C. The grid current will be linearly reduced by the "dc_breaker_derating" amount per 50°C above this point. Example: dc_breaker_derating is set to '50'. The grid current would be at maximum at 25°C, and reduced by 50% by the time the charger is at 75°C. Since the reduction is linear, it would be reduced by 25% at 50° in this example.	

11.7.3 ACUFlash Profile

This profile contains a set of keys that will appear to customers who are using Veefil units that are also equipped with an AC backpack. This profile pertains to an isolated embedded component in the Veefil unit, namely the ACU. A reboot of the ACU is required to enforce a change of any of these values. Therefore, these values can only be changed when the Veefil unit is not charging.

Key	R/RW	Type	Default	Specification description	Comments
HU1.ACUFlashDCMeter Multiplier	RW	Integer	1	Multiplier for DC meter read if installed	Previously *acu_dc_meter_multiplier*.
HU1.ACUFlashACMeter Multiplier	RW	Integer	1	Multiplier for AC meter read if installed	Previously *acu_ac_meter_multiplier*.

11.8 Advance Profiles - Grouped Chargers

For grouped chargers, the advanced configuration parameters must be specified per charger. For this reason, the configuration keys are specified on a per charger basis, and this done by prepending a token according to the convention "*HU{unitId}.{Key}*", where *unitId* is an integer starting from one, corresponding to distinct head unit; and *Key* is any configuration key from the advanced profiles listed above. For example, in a group system the remote end could supply the configuration key *HU1.CCURFIDDisable* to control the *CCURFIDDisable* setting on head unit 1, and similar could supply

HU2.CCUFlashDCMaxOutputPower to control *CCUFlashDCMaxOutputPower* on head unit 2, and so forth.

12 Revision record

Revision	Date	Change
1	1 December 2017	Document creation (TS)
2	12 December 2017	Added Grouped Chargers section (DW)
3	18 December 2017	Added Architectural Summary section with figures. Moved configuration section to the end. (DW)
4	10 January 2018	Minor updates to sections 11.7.1, 11.7.2, 11.7.3, and 11.8. (DW)