B. Provisioning

1. Introduction

This Functional Block describes all the functionalities that help a CSO provision their Charging Stations, allowing them on their network and retrieving configuration information from these Charging Stations. Additionally, it consists of the ability to retrieve information about the configuration of Charging Stations, make changes to the configuration etc. This chapter also covers resetting a Charging Station and migrating to a new NetworkConnectionProfile.

1.1. Transactions before being accepted by a CSMS

A Charging Station Operator MAY choose to configure a Charging Station to accept transactions before the Charging Station is accepted by a CSMS. Parties who want to implement this such behavior should realize that it is uncertain if those transactions can ever be delivered to the CSMS.

After a restart (for instance due to a remote reset command, power outage, firmware update, software error etc.) the Charging Station MUST again contact the CSMS and SHALL send a BootNotification request. If the Charging Station fails to receive a BootNotificationResponse from the CSMS, and has no in-built non-volatile real-time clock hardware that has been correctly preset, the Charging Station may not have a valid date and time setting, making it difficult or even impossible to later determine the date and time of transactions.

It might also be the case (e.g. due to configuration error) that the CSMS indicates a status other than Accepted for an extended period of time, or indefinitely.

It is usually advisable to deny all charging services at a Charging Station if the Charging Station has never before been Accepted by the CSMS (using the current connection settings, URL, etc.) since users cannot be authenticated and running transactions could conflict with provisioning processes.

If this is supported, this behaviour can be configured via the Configuration Variable: `TxBeforeAcceptedEnabled`.

2. Use cases & Requirements

2.1. Booting a Charging Station

B01 - Cold Boot Charging Station

Table 35. B01 - Cold Boot Charging Station

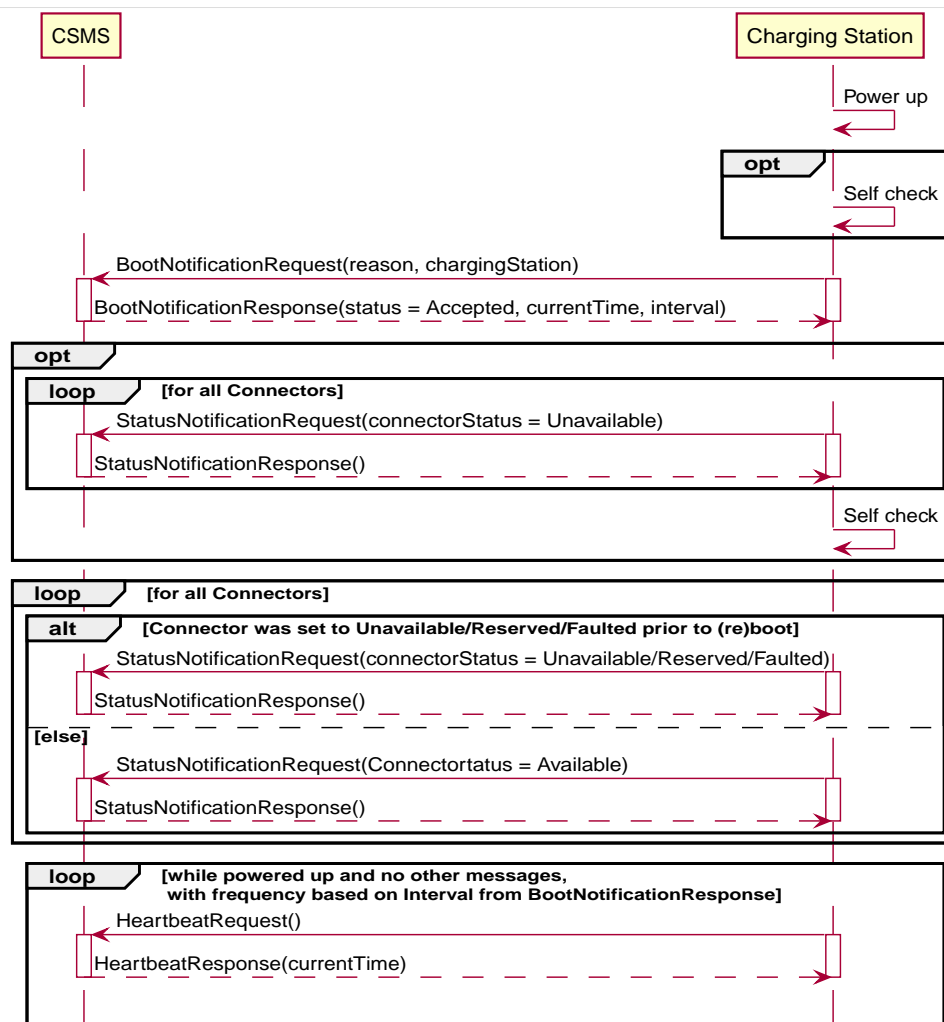| No.	Type	Description
1	**Name**	Cold Boot Charging Station
2	**ID**	B01
	Functional block	B. Provisioning
3	**Objective(s)**	The objective of this use case is to enable a Charging Station that is powering up to register itself at a CSMS and provide the right state information.
4	**Description**	This use case describes how the CSMS can control which Charging Stations access it. To be able to control Charging Stations connecting to a CSMS, Charging Stations are required to send BootNotificationRequest. This request contains some general information about the Charging Station.
	Actors	Charging Station, CSMS
	Scenario description	**1.** The Charging Station is powered up. **2.** The Charging Station sends BootNotificationRequest to the CSMS. **3.** The CSMS returns with BootNotificationResponse with the status *Accepted*. **4.** *Optional:* The Charging Station sends StatusNotificationRequest with status *Unavailable* to the CSMS for each Connector. **5.** The Charging Station sends StatusNotificationRequest to the CSMS for each Connector. If the status was set to *Unavailable* or *Reserved* from the CSMS prior to the (re)boot, the Connector should return to this status, otherwise the status should be *Available* or, when it resumes a transaction that was ongoing, the status should be *Occupied*. **6.** Normal operational is resumed. **7.** The Charging Station sends HeartbeatRequest to the CSMS.
	Alternative scenario(s)	B02 - Cold Boot Charging Station - Pending B03 - Cold Boot Charging Station - Rejected
5	**Prerequisite(s)**	The Charging Station is powered down.
6	**Postcondition(s)**	**Successful postcondition:** The Charging Station is in *Idle* status, and *Accepted*. **Failure postcondition:** The Charging Station received the status *Rejected*, B03 - Cold Boot Charging Station -Rejected applies. The Charging Station received the status *Pending*, B02 - Cold Boot Charging Station - Pending applies.

Figure 10. Sequence Diagram: Cold Boot Charging Station

7	Error handling	**1.** No initial establishment of connection of communication between the CSMS and Charging Station: Retry Connection with the CSMS. **2.** No response / time-out from the CSMS: The Charging Station resends BootNotificationRequest after a waiting interval. The Charging Station chooses this interval on its own (since it dit not get a BootNotificationResponse containing this interval), in a way that avoids flooding the CSMS with requests.
8	Remark(s)	Multiple options for a self check are possible: some Charging Stations boot and send status notifications with *Unavailable*, then perform a check of all the hardware and send new StatusNotifications with status *Available* when the Charging Station is up and running. However, there is no required order for a self check and sending a BootNotificationRequest. A Charging Stations can also do the self check *before* sending a BootNotificationRequest and determine the status before a (mobile) network connection is established and a BootNotificationRequest is sent. When something is wrong with the Charging Station or EVSE, the status SHALL be set to *Faulted*. *Reserved* and *Unavailable* states persist after a reboot.

B01 - Cold Boot Charging Station - Requirements

Table 36. B01 - Requirements

ID	Precondition	Requirement definition	Note
B01.FR.01	After start-up.	The Charging Station SHALL send BootNotificationRequest to the CSMS with information about its configuration.	Information: e.g. version, vendor, etc.
B01.FR.02	B01.FR.01 The CSMS has received BootNotificationRequest from the Charging Station.	The CSMS SHALL respond to indicate whether it will accept the Charging Station.	

ID	Precondition	Requirement definition	Note
B01.FR.03	After a reboot (for instance due to a remote reset command, power outage, firmware update, software error etc.)	The Charging Station SHALL again connect to the CSMS and SHALL send a BootNotificationRequest each time it boots or reboots.	
B01.FR.04	When the CSMS responds with BootNotificationResponse with the status *Accepted*.	The Charging Station SHALL adjust the heartbeat interval in accordance with the interval from the response message.	
B01.FR.05	When the CSMS responds with BootNotificationResponse with the status *Accepted*.	The Charging Station SHALL send a StatusNotificationRequest for each Connector with its current state.	
B01.FR.06	The Charging Station has received BootNotificationResponse. AND Charging Station is configured to use Heartbeats for time synchronization `TimeSource`	The Charging Station SHALL synchronize the Charging Station's internal clock with the supplied CSMS's current time.	
B01.FR.07	When a Charging Station or an EVSE is set to status *Unavailable* by a Change Availability command.	The *Unavailable* status MUST be persistent across reboots.	
B01.FR.08	Between the physical power-on/reboot and the successful completion of a BootNotification, where the CSMS returns *Accepted* or *Pending*.	The Charging Station SHALL NOT send any other OCPP requests to the CSMS (Except BootNotificationRequest). This includes cached OCPP messages that are still present in the Charging Station from prior sessions.	Refer to B02 - Cold Boot Charging Station - Pending (for example B02.FR.02) for more details on sending messages on the *Pending* status.
B01.FR.09	B01.FR.01	The Charging Station SHALL indicate the reason for sending the BootNotificationRequest message in the *reason* field.	For which reason to use, see BootReasonEnumType.
B01.FR.10	The Charging Station has received a BootNotificationResponse in which status is not *Accepted* AND the Charging Station sends a RPC Framework: CALL message that is NOT a BootNotificationRequest or a message triggered by one of the following messages: TriggerMessageRequest, GetBaseReportRequest, GetReportRequest.	The CSMS SHALL respond with RPC Framework: CALLERROR: SecurityError.	The Charging Station is not allowed to initiate sending other messages before being accepted.
B01.FR.11	B01.FR.01 AND Security profile 3 is used	The CSMS SHALL check the SerialNumber in the BootNotificationRequest against the Serial Number in the Certificate Common Name.	
B01.FR.12	B01.FR.11 AND the SerialNumber in the BootNotificationRequest does NOT equal the Serial Number in the Certificate Common Name	The CSMS SHALL close WebSocket connection.	
B01.FR.13	When an EVSE has been reserved	The *Reserved* state MUST be persistent across reboots.	

B02 - Cold Boot Charging Station - Pending

Table 37. B02 - Cold Boot Charging Station - Pending

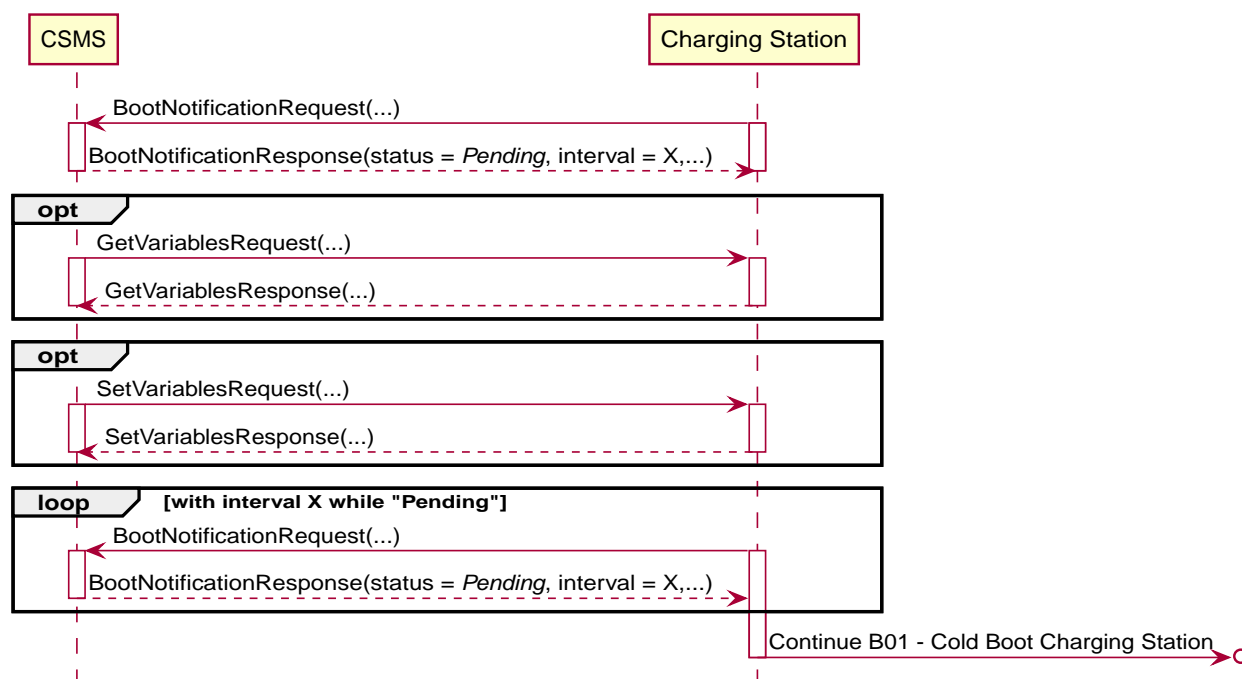| No.	Type	Description
1	Name	Cold Boot Charging Station - Pending
2	ID	B02
	Functional block	B. Provisioning
	Parent use case	B01 - Cold Boot Charging Station
3	Objective(s)	**1.** To inform the Charging Station that it is not yet accepted by the CSMS: *Pending* status.
		2. To give the CSMS a way to retrieve or set certain configuration information. **3.** To give the CSMS a way of limiting the load on the CSMS after e.g. a reboot of the CSMS.
4	Description	This use case describes the behavior of the CSMS and a Charging Station when the Charging Station is informed by the CSMS that it is not yet accepted using the *Pending* status.
	Actors	Charging Station, CSMS
	Scenario description	**1.** The Charging Station is powered up.
		2. The Charging Station sends BootNotificationRequest to the CSMS.
		3. The CSMS responds with BootNotificationResponse with the status *Pending*. **4.** The CSMS then, is able to send messages to the Charging Station in order to change the configuration of the Charging Station. **5.** The Charging Station resends BootNotificationRequest after the number of seconds indicated by the interval field. (Interval from BootNotificationResponse)
5	Prerequisite(s)	**1.** The CSMS requires to set the Charging Station in *Pending* status. **2.** The Charging Station is starting up (i.e. powering up after being powered down).
6	Postcondition(s)	**Successful postcondition:** The Charging Station is in *Pending* status. **Failure postcondition:** The Charging Station received the status *Rejected*, B03 - Cold Boot Charging Station -Rejected applies.

Figure 11. Sequence Diagram: Cold Boot Charging Station - Pending

7	Error handling	**1.** When no initial connection established between CSMS and Charging Station: Retry Connection to the CSMS and resend BootNotificationRequest. **2.** No response / time-out from the CSMS: The Charging Station resends BootNotificationRequest after a waiting interval. This waiting interval can be based on the interval from a previous BootNotificationResponse or chosen by the Charging Station itself. In the latter case, the Charging Station chooses this interval in a way that avoids flooding the CSMS with requests.
8	Remark(s)	When the CSMS returns with BootNotificationResponse with the status *Accepted*, B01 - Cold Boot Charging Station applies.

B02 - Cold Boot Charging Station - Pending - Requirements

Table 38. B02 - Requirements

ID	Precondition	Requirement definition	Note
B02.FR.01	After the Charging Station received the *Pending* status.	The CSMS MAY send messages to retrieve information from the Charging Station (as described in use cases B06, B07, B08) or change its configuration by SetVariablesRequest (as described in use case B05). The Charging Station SHALL respond to these messages.	The Pending status can thus indicate that the CSMS wants to retrieve or set certain information on the Charging Station before it will accept the Charging Station.
B02.FR.02	While the CSMS has not yet responded to a BootNotificationRequest with an *Accepted* status in the BootNotificationResponse.	The Charging Station SHALL NOT send RPC Framework: CALL messages (Except BootNotificationRequest) to the CSMS, unless it has been instructed by the CSMS to do so, using one of the following messages: TriggerMessageRequest, GetBaseReportRequest, GetReportRequest.	
B02.FR.03	While the CSMS has not yet responded to a BootNotificationRequest with an *Accepted* status in the BootNotificationResponse.	A Charging Station Operator MAY choose to configure a Charging Station to accept transactions and queue TransactionEventRequest messages to be sent to the CSMS	Parties who want to implement this behavior must realize that it is uncertain if those transactions can ever be delivered to the CSMS.
B02.FR.04	While the CSMS has not yet responded to a BootNotificationRequest with an *Accepted* status in the BootNotificationResponse.	A Charging Station SHALL NOT send BootNotificationRequest earlier than the value of the Interval field in BootNotificationResponse, unless requested to do so with TriggerMessageRequest.	
B02.FR.05	While in *Pending* status AND receiving a RequestStartTransactionRequest or RequestStopTransactionRequest	The Charging Station SHALL respond with a RequestStartTransactionResponse or RequestStopTransactionResponse with status *Rejected*. (Even if the Charging Station is allowed to start transaction, see B02.FR.03. If the CSMS wants to use RequestStartTransaction etc. it SHALL first accept the Charging Station)	
B02.FR.06	When the CSMS returns the Pending status	The communication channel SHALL NOT be closed by either the Charging Station or the CSMS.	
B02.FR.07	If the interval in the BootNotificationResponse equals 0, and the status is other than *Accepted*,	The Charging Station SHALL choose a waiting interval on its own, in a way that avoids flooding the CSMS with requests.	
B02.FR.08	If the interval in the BootNotificationResponse > 0, and the status is other than *Accepted*,	The Charging Station SHALL send a BootNotificationRequest after the set interval has past.	

ID	Precondition	Requirement definition	Note
B02.FR.09	The Charging Station has received a BootNotificationResponse with status *Pending* AND the Charging Station sends a RPC Framework: CALL message that is NOT a BootNotificationRequest or a message triggered by one of the following messages: TriggerMessageRequest, GetBaseReportRequest, GetReportRequest.	The CSMS SHALL respond with RPC Framework: CALLERROR: SecurityError.	The Charging Station is not allowed to initiate sending other messages before being accepted.

B03 - Cold Boot Charging Station - Rejected

Table 39. B03 - Cold Boot Charging Station - Rejected

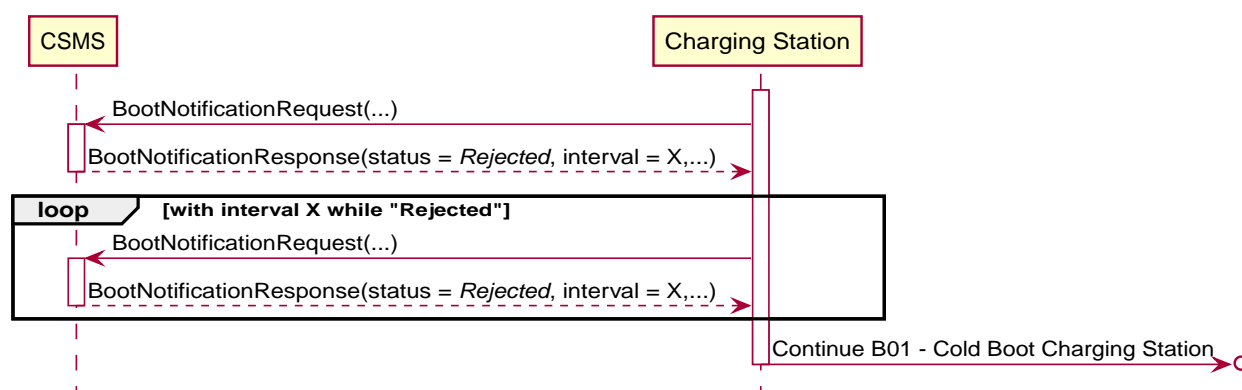| No.	Type	Description
1	Name	Cold Boot Charging Station - Rejected
2	ID	B03
	Functional block	B. Provisioning
	Parent use case	B01 - Cold Boot Charging Station
3	Objective(s)	To inform the Charging Station that its *not* (yet) accepted by the CSMS: *Rejected* status.
4	Description	This use case describes the behavior of the CSMS and a Charging Station, when the Charging Station is informed by the CSMS that it is not (yet) accepted using the *Rejected* status.
	Actors	Charging Station, CSMS
	Scenario description	**1.** The Charging Station is powered up. **2.** The Charging Station sends BootNotificationRequest to the CSMS. **3** The CSMS responds with BootNotificationResponse with the status *Rejected* to the Charging Station. **4.** The Charging Station will resend BootNotificationRequest after the number of seconds indicated by the interval field. (Interval from BootNotificationResponse).
5	Prerequisite(s)	**1.** The CSMS requires to set the Charging Station in the *Rejected* status. **2.** The Charging Station is powered down.
6	Postcondition(s)	The Charging Station remains in the *Rejected* status.

Figure 12. Sequence Diagram: Cold Boot Charging Station - Rejected

7	Error handling	When there is no response or a time-out from the CSMS: The Charging Station resends BootNotificationRequest after a waiting interval. This waiting interval can be based on the interval from a previous BootNotificationResponse or chosen by the Charging Station itself. In the latter case, the Charging Station chooses this interval in a way that avoids flooding the CSMS with requests.
8	Remark(s)	During the status *Rejected*, the Charging Station may no longer be reachable from the CSMS. The Charging Station MAY e.g. close its communication channel or shut down its communication hardware. Additionally, the CSMS MAY close the communication channel, for instance to free up system resources. It is advised *not* to accept any transactions until the BootNotification of the Charging Station has been accepted by the CSMS. See: Transactions before being accepted by a CSMS When the CSMS returns with BootNotificationResponse with the status *Accepted*, B01 - Cold Boot Charging Station applies.

B03 - Cold Boot Charging Station - Rejected - Requirements

Table 40. B03 - Requirements

ID	Precondition	Requirement definition
B03.FR.01	If the Charging Station is configured to accept Transactions before being accepted by a CSMS	The Charging Station MAY allow locally authorized transactions.
B03.FR.02	If the CSMS returns the status *Rejected*. For example when a Charging Station is blacklisted.	The Charging Station SHALL NOT send any OCPP message to the CSMS until the retry interval has expired.
B03.FR.03	While in the status *Rejected*.	The CSMS SHALL NOT initiate any messages.
B03.FR.04	B03.FR.03	The Charging Station MAY close the connection until it needs to send the next BootNotificationRequest.
B03.FR.05	If the interval in the BootNotificationResponse equals 0, and the status is other than *Accepted*	The Charging Station SHALL choose a waiting interval on its own, in a way that avoids flooding the CSMS with requests.
B03.FR.06	If the interval in the BootNotificationResponse is greater than 0, and the status is other than *Accepted*	The Charging Station SHALL send a BootNotificationRequest after the set interval has past.
B03.FR.07	B03.FR.03 AND Charging Station sends a message that is not a BootNotificationRequest	CSMS SHALL respond with RPC Framework: CALLERROR: SecurityError.
B03.FR.08	B03.FR.03 AND CSMS sends a message that is not a TriggerMessageRequest(requestedMessage = *BootNotification*)	Charging Station SHALL respond with RPC Framework: CALLERROR: SecurityError.

B04 - Offline Behavior Idle Charging Station

Table 41. B04 - Offline Behavior Idle Charging Station

| No.	Type	Description
1	**Name**	Offline Behavior Idle Charging Station
2	**ID**	B04
	Functional block	B. Provisioning
3	**Objective(s)**	To attain stand-alone operation of the Charging Station.
4	**Description**	This use case describes that, in the event of unavailability of the communication, the Charging Station is designed to operate stand-alone. In that situation, the Charging Station is said to be *Offline*.
	Actors	Charging Station, CSMS
	Scenario description	**1.** The CSMS or communication is unavailable. **2.** The Charging Station operates stand-alone. **3.** The connection is restored. **4.** If the *Offline* period exceeds the value of the OfflineThreshold Configuration Variable: the Charging Station sends a StatusNotificationRequest to the CSMS for each connector. Otherwise it only sends a StatusNotificationRequest for Connectors with a status change during the offline period. **5.** The Charging Station sends HeartbeatRequest to the CSMS. **6.** The CSMS responds with HeartbeatResponse.
5	**Prerequisite(s)**	The BootNotification was previously accepted and the Charging Station is able to operate stand-alone.
6	**Postcondition(s)**	When connection is restored after a period of *Offline* behavior, the CSMS knows the Charging Stations' and EVSEs' state.
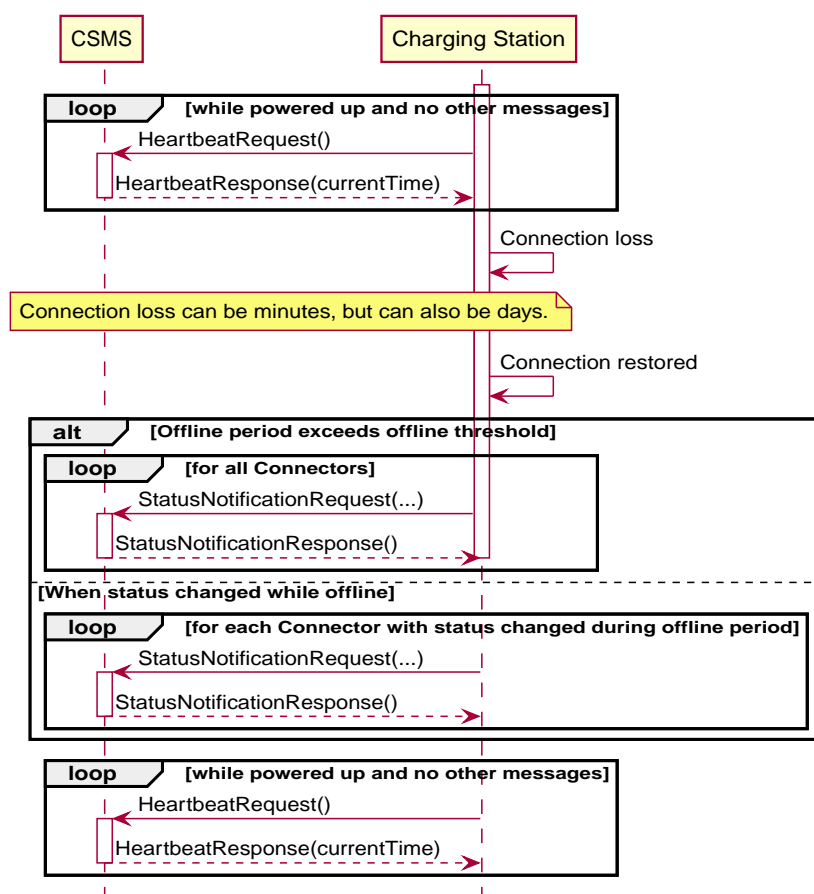
Figure 13. Sequence Diagram: Offline Behavior Idle Charging Station

7	**Error handling**	The offline situation is an non preferred mode of operation that needs to be handled by the Charging Station by trying to re-establish the connection.
8	**Remark(s)**	n/a

B04 - Offline Behavior Idle Charging Station - Requirements

Table 42. B04 - Requirements

ID	Precondition	Requirement definition
B04.FR.01	After having been *Offline* AND the *Offline* period exceeds the value of the OfflineThreshold Configuration Variable.	The Charging Station SHALL send StatusNotificationRequest to report the current status of all its Connectors.
B04.FR.02	After having been *Offline* AND the *Offline* period does NOT exceed the value of the OfflineThreshold Configuration Variable.	The Charging Station SHALL send StatusNotificationRequest to report the current status of only the Connectors for which a state change occurred.

2.2. Configuring a Charging Station

NOTE	For managing the configuration of a Charging Station a basic understanding of Device Model concepts is essential. These concepts are explained in "OCPP 2.0.1: Part 1 - Architecture & Topology", chapter 4.

B05 - Set Variables

Table 43. B05 - Set Variables

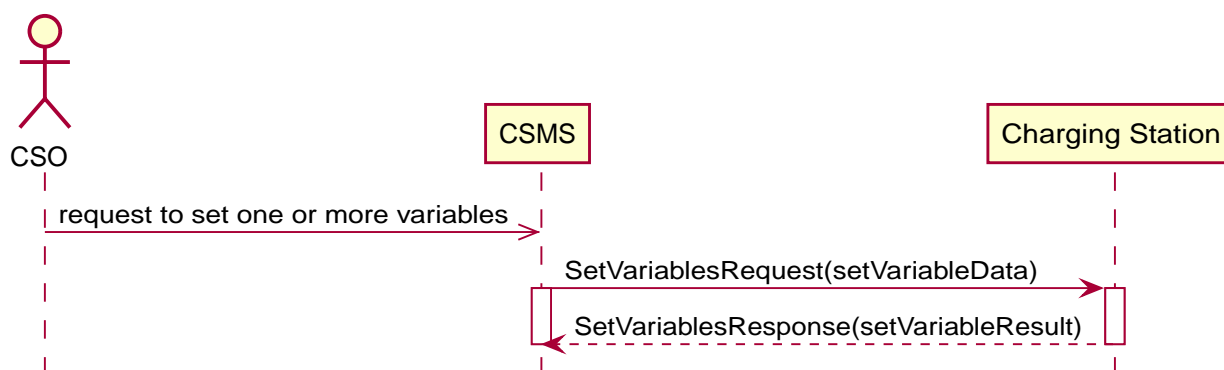| No.	Type	Description
1	**Name**	Set Variables
2	**ID**	B05
	Functional block	B. Provisioning
3	**Objective(s)**	To give the CSMS the ability to make changes to variables in the Charging Station.
4	**Description**	A Charging Station can have a lot of variables that can be configured/changed by the CSMS. A CSMS can use these variables to for example influence the behavior of a Charging Station. This use case describes how the CSMS requests a Charging Station to set the value of variables of a component. The CSMS can request to set more than one value per request.
	Actors	CSMS, Charging Station
	Scenario description	**1.** The CSO triggers the CSMS to request setting one or more variables in a Charging Station. **2.** The CSMS sends a SetVariablesRequest to the Charging Station. **3.** The Charging Station responds with a SetVariablesResponse indicating whether it was able to executed the change(s).
5	**Prerequisite(s)**	n/a
6	**Postcondition(s)**	**Successful postconditions:** **1.** The change was executed *Successfully*. **Failure postconditions:** **1.** The variable is supported, but setting could not be changed, the Charging Station responds with the status *Rejected*. **2.** The variable is *not* supported, the Charging Station responds with the status *UnknownVariable*.

Figure 14. Sequence Diagram: Set Variables

7	**Error handling**	n/a

8	Remark(s)	The attributeType Actual corresponds with the actual value of the Variable, whereas the attributeTypes Target, MinSet and MaxSet correspond to the target, minimum and maximum values that have been set for this variable.
		This is best explained by an example: the cooling system is configured to operate with a fan speed between 1000 and 5000 rpm. These boundaries are represented by the MinSet and MaxSet attributes. The current fan speed is represented by the Actual attribute. The desired fan speed is represented by the Target attribute.

B05 - Set Variables - Requirements

Table 44. B05 - Requirements

ID	Precondition	Requirement definition
B05.FR.01	When the Charging Station receives a SetVariablesRequest with an X number of SetVariableData elements	The Charging Station SHALL respond with an SetVariablesResponse with an equal (X) number of SetVariableResult elements, one for every SetVariableData element in the SetVariablesRequest.
B05.FR.02	B05.FR.01	Every SetVariableResult element in the SetVariablesResponse SHALL contain the same *component* and *variable* combination as one of the SetVariableData elements in the SetVariablesRequest.
B05.FR.03	B05.FR.02 AND If the SetVariablesRequest contains an *attributeType*	The corresponding SetVariableResult element in the SetVariablesResponse SHALL also contain the same *attributeType*
B05.FR.04	When the Charging Station receives a SetVariablesRequest with an unknown Component in the SetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding SetVariableResult to: UnknownComponent.
B05.FR.05	When the Charging Station receives a SetVariablesRequest with a Variable that is unknown for the given Component in the SetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding SetVariableResult to: UnknownVariable.
B05.FR.06	When the Charging Station receives a SetVariablesRequest with an attributeType that is unknown for the given Variable in the SetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding SetVariableResult to: NotSupportedAttributeType.
B05.FR.07	When the Charging Station receives a SetVariablesRequest with a *value* that is incorrectly formatted for the given Variable in the SetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding SetVariableResult to: Rejected. (More information can be provided in the optional *statusInfo* element.)
B05.FR.08	When the Charging Station receives a SetVariablesRequest with a *value* that is lower or higher than the range of the given Variable in the SetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding SetVariableResult to: Rejected. (More information can be provided in the optional *statusInfo* element.)
B05.FR.09	NOT (B05.FR.04 to B05.FR.08) AND When the Charging Station receives a SetVariablesRequest for a Variable in the SetVariableData, but is not able to set it	The Charging Station SHALL set the *attributeStatus* field in the corresponding SetVariableResult to: Rejected. (This happens if the variable is *ReadOnly*, but may also occur when setting the variable fails because of technical problems.)
B05.FR.10	When the Charging Station was able to set the given *value* from the SetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding SetVariableResult to: Accepted.
B05.FR.11		The CSMS SHALL NOT send more SetVariableData elements in a SetVariablesRequest than reported by the Charging Station via `ItemsPerMessageSetVariables`.
B05.FR.12	When the Charging Station receives a SetVariablesRequest without an *attributeType*.	The corresponding SetVariableResult element in the SetVariablesResponse SHALL contain the *attributeType* Actual.
B05.FR.13		The CSMS SHALL NOT include multiple SetVariableData elements, in a single SetVariablesRequest, with the same Component, Variable and *AttributeType* combination. Note that an omitted *AttributeType* counts as the value *Actual*.

B06 - Get Variables

Table 45. B06 - Get Variables

| No.	Type	Description
1	Name	Get Variables
2	ID	B06
	Functional block	B. Provisioning
3	Objective(s)	To give the CSMS the ability to retrieve the value of an attribute for one or more Variables of one or more Components.
4	Description	This use case describes how the CSMS requests a Charging Station to send the value of an attribute for one or more variables of one or more components. It is not possible to get all attributes of all variables in one call.
	Actors	Charging Station, CSMS
	Scenario description	**1.** The CSO triggers the CSMS to request for a number of variables in a Charging Station. **2.** The CSMS request the Charging Station for a number of variables with GetVariablesRequest with a list of requested variables. **3.** The Charging Station responds with a GetVariablesResponse with the requested variables. **4.** The CSMS sends an optional notification to the CSO.
5	Prerequisite(s)	n/a
6	Postcondition(s)	**Successful postcondition:** The Charging Station was able to send all the requested variables. **Failure postcondition:** The Charging Station was not able to send all requested variables.
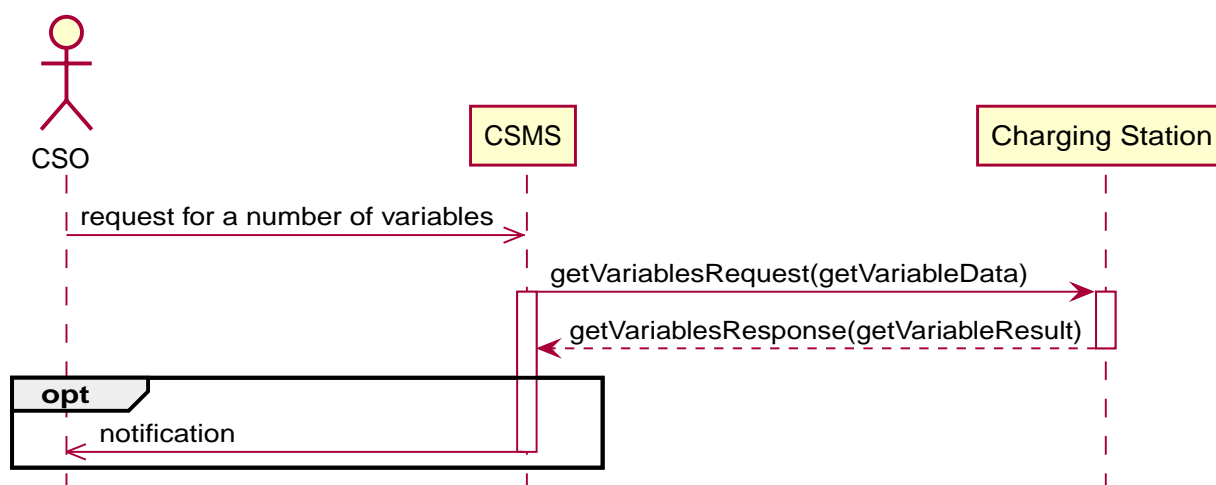
Figure 15. Sequence Diagram: Get Variables

7	Error handling	n/a
8	Remark(s)	n/a

B06 - Get Variables - Requirements

Table 46. B06 - Requirements

ID	Precondition	Requirement definition
B06.FR.01	When the Charging Station receives a GetVariablesRequest with an X number of GetVariableData elements	The Charging Station SHALL respond with an GetVariablesResponse with an equal (X) number of GetVariableResult elements, one for every GetVariableData element in the GetVariablesRequest.
B06.FR.02	B06.FR.01	Every GetVariableResult element in the GetVariablesResponse SHALL contain the same *component* and *variable* combination as one of the GetVariableData elements in the GetVariablesRequest.

ID	Precondition	Requirement definition
B06.FR.03	B06.FR.02 AND If the GetVariablesRequest contains an *attributeType*	The corresponding GetVariableResult element in the GetVariablesResponse SHALL also contain the same *attributeType*
B06.FR.04	B06.FR.01	Every GetVariableResult element in the GetVariablesResponse SHALL contain an *attributeValue* with the value of an attribute from the requested *attributeType* in the GetVariablesRequest.
B06.FR.05		The CSMS SHALL NOT send more GetVariableData elements in a GetVariablesRequest than reported by the Charging Station via `ItemsPerMessageGetVariables`.
B06.FR.06	When the Charging Station receives a GetVariablesRequest with an unknown Component in the GetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding GetVariableResult to: UnknownComponent AND SHALL omit the *attributeValue*.
B06.FR.07	When the Charging Station receives a GetVariablesRequest with a Variable that is unknown for the given Component in the GetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding GetVariableResult to: UnknownVariable AND SHALL omit the *attributeValue*.
B06.FR.08	When the Charging Station receives a GetVariablesRequest with an attributeType that is unknown for the given Variable in the GetVariableData	The Charging Station SHALL set the *attributeStatus* field in the corresponding GetVariableResult to: NotSupportedAttributeType AND SHALL omit the *attributeValue*.
B06.FR.09	When the Charging Station receives a GetVariablesRequest for a Variable in the GetVariableData that is *WriteOnly*	The Charging Station SHALL set the *attributeStatus* field in the corresponding GetVariableResult to: Rejected.
B06.FR.10	When the Charging Station was able to get the *value* requested from a GetVariablesRequest	The Charging Station SHALL set the *attributeStatus* field in the corresponding GetVariableResult to: Accepted and set the *attributeValue* to the found value.
B06.FR.11	When the Charging Station receives a GetVariablesRequest without an *attributeType*.	The corresponding GetVariableResult element in the GetVariablesResponse SHALL contain the *attributeType* Actual.
B06.FR.13	NOT B06.FR.08 AND the Charging Station has no *attributeValue* for the requested *attributeType* of the componentvariable	Charging Station SHALL return an empty string as *attributeValue*. Note: this can happen, for example, when the *attributeType* `Target` has not yet been set, even though it is supported.

B07 - Get Base Report

Table 47. B07 - Get Base Report

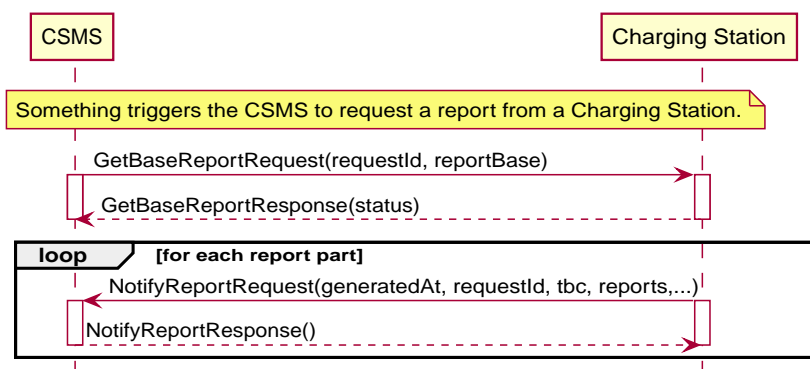| No.	Type	Description
1	**Name**	Get Base Report
2	**ID**	B07
	Functional block	B. Provisioning
3	**Objective(s)**	To give the CSMS the ability to request a predefined report as defined in ReportBase.
4	**Description**	This use case describes how the CSMS requests a Charging Station to send a predefined report as defined in ReportBase. The result will be returned asynchronously in one or more NotifyReportRequest messages.
	Actors	Charging Station, CSMS
	Scenario description	**1.** The CSO triggers the CSMS to request a report from a Charging Station. **2.** The CSMS requests the Charging Station for a report with GetBaseReportRequest. **3.** The Charging Station responds with GetBaseReportResponse. **4.** The Charging Station asynchronously sends the results in one or more NotifyReportRequest messages. **5.** The CSMS responds with NotifyReportResponse for each NotifyReportRequest.
5	**Prerequisite(s)**	n/a
6	**Postcondition(s)**	**Successful postcondition:** The Charging Station was able to send the requested report. **Failure postcondition:** The Charging Station was *not* able to send the requested report.

Figure 16. Sequence Diagram: Get Base Report

7	**Error handling**	n/a
8	**Remark(s)**	n/a

B07 - Get Base Report - Requirements

Table 48. B07 - Requirements

ID	Precondition	Requirement definition	Note
B07.FR.01	When the Charging Station receives a getBaseReportRequest for a supported *reportBase* AND NOT B07.FR.13	The Charging Station SHALL send a getBaseReportResponse with `Accepted`.	
B07.FR.02	When the Charging Station receives a getBaseReportRequest for a *reportBase* that is not supported	The Charging Station SHALL send a getBaseReportResponse with `NotSupported`.	

ID	Precondition	Requirement definition	Note
B07.FR.03	B07.FR.01	The Charging Station SHALL send the requested information via one or more NotifyReportRequest messages to the CSMS.	
B07.FR.04	B07.FR.01 AND The getBaseReportRequest contained a *requestId*	Every NotifyReportRequest send for this getBaseReportRequest SHALL contain the same *requestId*.	
B07.FR.05	B07.FR.02	The Charging Station SHALL NOT send a NotifyReportRequest to the CSMS.	
B07.FR.07	B07.FR.01 AND When *reportBase* is ConfigurationInventory	Then the Charging Station SHALL respond with a NotifyReportRequest to report on all component-variables that can be set by the operator including their *VariableCharacteristics*.	
B07.FR.08	B07.FR.01 AND When *reportBase* is FullInventory	Then the Charging Station SHALL respond with a NotifyReportRequest to report on all component-variables including their *VariableCharacteristics*.	As a minimum the required variables mentioned in Charging Infrastructure related shall be reported as well as the required variables in Section 1 Controller Components that are relevant to each functional block that has been implemented.
B07.FR.09	B07.FR.01 AND When *reportBase* is SummaryInventory	Then the Charging Station SHALL respond with a NotifyReportRequest to report on components and variables related to the availability and condition of the Charging Station, notably operationalStatus of the Charging Station, EVSE and Connectors and any error condition.	A (summary) report that lists Components/Variables relating to the Charging Station's current charging availability, and to any existing problem conditions. For the Charging Station Component: - AvailabilityState. For each EVSE Component: - AvailabilityState. For each Connector Component: - AvailabilityState (if known and different from EVSE). For all Components in an abnormal State: - Active (Problem, Tripped, Overload, Fallback) variables. - Any other diagnostically relevant Variables of the Components.
B07.FR.10		The sequence number contained in the seqNo field of the NotifyReportRequest is incremental per report. So the NotifyReportRequest message which contains the first report part, SHALL have a seqNo with value *0*.	
B07.FR.11	B07.FR.08	All attribute types of a variable, that are supported by the Charging Station, SHALL be reported, even if they have no value (are unset).	This allows a CSMS to know which attribute types are supported by the Charging Station.
B07.FR.12		The Charging Station SHALL support at least the base reports: ConfigurationInventory and FullInventory.	
B07.FR.13	When the Charging Station is temporarily unable to execute a report request	The Charging Station SHALL send a getBaseReportResponse with `Rejected`.	

B08 - Get Custom Report

Table 49. B08 - Get Custom Report

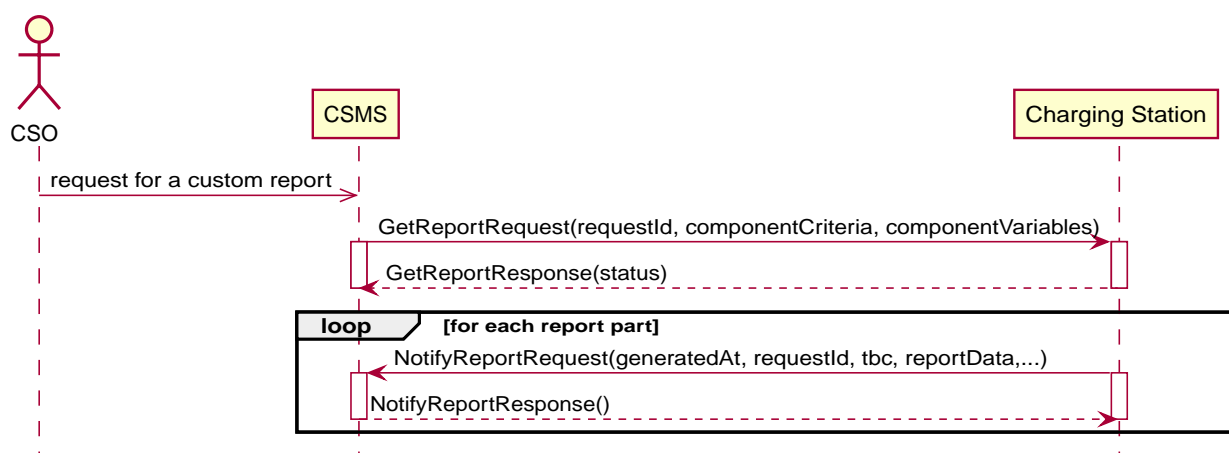| No.	Type	Description
1	**Name**	Get Custom Report
2	**ID**	B08
	Functional block	B. Provisioning
3	**Objective(s)**	To give the CSMS the ability to request a report of all Components and Variables limited to those that match ComponentCriteria and/or the list of ComponentVariables.
4	**Description**	This use case describes how the CSMS requests a Charging Station to send a report of all Components and Variables limited to those that match ComponentCriteria and/or the list of ComponentVariables. The result will be returned asynchronously in one or more NotifyReportRequest messages.
	Actors	Charging Station, CSMS
	Scenario description	**1.** The CSO triggers the CSMS to request a report from a Charging Station. **2.** The CSMS requests the Charging Station for a report with a GetReportRequest. **3.** The Charging Station responds with a GetReportResponse. **4.** The Charging Station asynchronously sends the results in one or more NotifyReportRequest messages. **5.** The CSMS responds with a NotifyReportResponse.
5	**Prerequisite(s)**	n/a
6	**Postcondition(s)**	**Successful postcondition:** The Charging Station was able to send the requested report. **Failure postcondition:** The Charging Station was *not* able to send the requested report.

Figure 17. Sequence Diagram: Get Custom Report

7	**Error handling**	n/a
8	**Remark(s)**	n/a

B08 - Get Custom Report - Requirements

Table 50. B08 - Requirements

ID	Precondition	Requirement definition
B08.FR.01	When the Charging Station receives a getReportRequest for supported *criteria*	The Charging Station SHALL send a getReportResponse with Accepted
B08.FR.02	When the Charging Station receives a getReportRequest for not supported *criteria*	The Charging Station SHALL send a getReportResponse with NotSupported
B08.FR.03	B08.FR.01	The Charging Station SHALL send the requested information via one or more NotifyReportRequest messages to the CSMS.

ID	Precondition	Requirement definition
B08.FR.04	B08.FR.01 AND The getReportRequest contained a *requestId*	Every NotifyReportRequest sent for this getReportRequest SHALL contain the same *requestId*.
B08.FR.05	B08.FR.01 AND *componentCriteria* and *componentVariables* are NOT both empty.	Every NotifyReportRequest sent for this getReportRequest SHALL be limited to the set *componentCriteria* and *componentVariables*.
B08.FR.06		The maximum number of *componentVariables* in one getReportRequest message is given by the `ItemsPerMessageGetReport` Configuration Variable
B08.FR.07	B08.FR.01 AND *ComponentCriteria* contains: *Active*	The Charging Station SHALL report every component that has the variable *Active* set to *true*, or does not have the *Active* variable in a NotifyReportRequest.
B08.FR.08	B08.FR.01 AND *ComponentCriteria* contains: *Available*	The Charging Station SHALL report every component that has the variable *Available* set to *true*, or does not have the *Available* variable, in a NotifyReportRequest.
B08.FR.09	B08.FR.01 AND *ComponentCriteria* contains: *Enabled*	The Charging Station SHALL report every component that has the variable *Enabled* set to *true*, or does not have the *Enabled* variable, in a NotifyReportRequest.
B08.FR.10	B08.FR.01 AND *ComponentCriteria* contains: *Problem*	The Charging Station SHALL report every component that has the variable *Problem* set to *true* in a NotifyReportRequest.
B08.FR.11	B08.FR.01 AND *componentCriteria* is absent AND *componentVariables* is NOT empty.	Every NotifyReportRequest sent for this getReportRequest is limited to the set in *componentVariables*.
B08.FR.12	B08.FR.01	The reported variables in NotifyReportRequest SHALL contain *variableCharacteristics*.
B08.FR.13	B08.FR.01 AND More than one *componentCriteria* is given.	The Charging Station SHALL report all components that have at least one of the given criteria (logical OR).
B08.FR.14		The sequence number contained in the seqNo field of the NotifyReportRequest is incremental per report. So the NotifyReportRequest message which contains the first report part, SHALL have a seqNo with value *0*.
B08.FR.15	When the Charging Station receives a GetReportRequest with a combination of criteria which results in an empty result set.	The Charging Station SHALL respond with a GetReportResponse(*status*=`EmptyResultSet`).
B08.FR.16	When the Charging Station is temporarily unable to execute a report request	The Charging Station SHALL send a getBaseReportResponse with `Rejected`.

B09 - Setting a new NetworkConnectionProfile

Table 51. B09 - Setting a new NetworkConnectionProfile

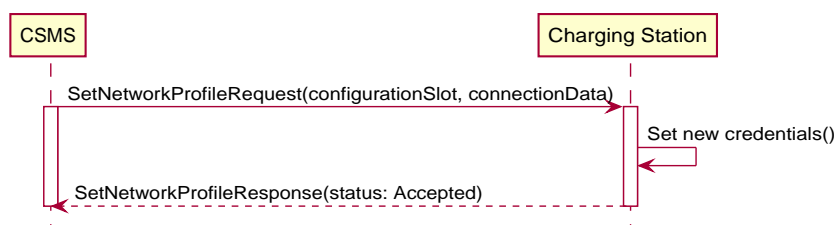| No.	Type	Description
1	**Name**	Setting a new NetworkConnectionProfile.
2	**ID**	B09
	Functional block	B. Provisioning
3	**Objectives**	To enable the CSMS to update the connection details on the Charging Station.
4	**Description**	The CSMS updates the connection details on the Charging Station. For instance in preparation of a migration to a new CSMS. After completion of this use case, the Charging Station to CSMS connection data has been updated.
	Actors	Charging Station, CSMS
	Scenario description	**1.** The CSMS sends a SetNetworkProfileRequest PDU containing an updated connection profile **2.** The Charging Station receives the PDU, validates the content and stores the new data **3.** The Charging Station responds by sending a SetNetworkProfileResponse PDU, with status *Accepted*
5	**Prerequisites**	The data supplied by the CSMS matches the Charging Station's capabilities
6	**Postcondition(s)**	The Charging Station was able to store the new connection data

Figure 18. Sequence Diagram: Set Network Connection Profile

8	**Error Handling**	Activation of a new NetworkConnectionProfile is described in B10 - Migrate to new CSMS. Errors during this use-case are not destructive to the current data connection. Error handling is further described in B10 - Migrate to new CSMS
9	**Remarks**	Even when changes are made to the currenctly active NetworkConnectionProfile, these will not activated until a reboot has occured, as described in B10 - Migrate to new CSMS.

B09 - Setting a new NetworkConnectionProfile - Requirements

Table 52. B09 - Requirements

ID	Precondition	Requirement definition
B09.FR.01	On receipt of the SetNetworkProfileRequest	The Charging Station SHALL validate the content, store the new data and if successful, respond by sending a SetNetworkProfileResponse message, with status *Accepted*
B09.FR.02	On receipt of the SetNetworkProfileRequest	The Charging Station SHALL validate the content. If the content is invalid, the Charging Station SHALL respond by sending a SetNetworkProfileResponse message, with status *Rejected*
B09.FR.03	If setting the new networkprofile fails.	The Charging Station SHALL respond by sending a SetNetworkProfileResponse message, with status *Failed*
B09.FR.04	On receipt of the SetNetworkProfileRequest AND the NetworkConnectionProfile contains a lower securityProfile than stored at the configuration variable `SecurityProfile`	The Charging Station SHALL respond by sending a SetNetworkProfileResponse message, with status *Rejected*

B10 - Migrate to new CSMS

Table 53. B10 - Migrate to new CSMS

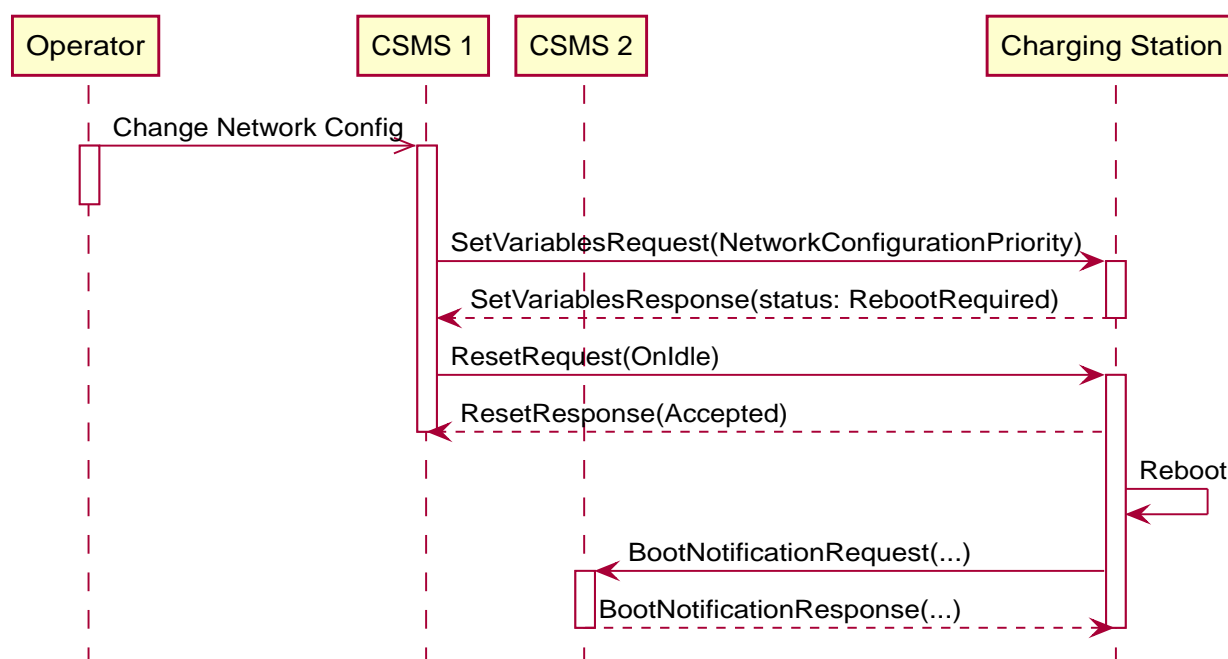| No.	Type	Description
1	**Name**	Migrate to new CSMS, using a different NetworkConnectionProfile.
2	**ID**	B10
	Functional block	B. Provisioning
3	**Objectives**	After completion of this use case, the Charging Station connects to a new CSMS.
4	**Description**	This use case describes how a Charging Station can be instructed to connect to a new CSMS, by changing the order of NetworkConnectionProfiles in NetworkConfigurationPriority.
	Actors	Charging Station, CSMS 1, CSMS 2
	Scenario description	**1.** CSMS 1 sets a new value for the NetworkConfigurationPriority Configuration Variable via SetVariablesRequest, such that the NetworkConnectionProfile for CSMS 2 becomes first in the list and the existing connection to CSMS 1 becomes second in the list. **2.** The Charging Station responds with a SetVariablesResponse with status *Accepted* **3.** CSMS 1 instructs the Charging Station to perform a `Reset OnIdle`. **4.** The Charging Station reboots and connects via the new primary NetworkConnectionProfile to CSMS 2.
5	**Prerequisites**	Use case B09 - Setting a new NetworkConnectionProfile was executed successfully prior to this use case The data supplied by the CSMS matches the Charging Station's capabilities
6	**Postcondition(s)**	The Charging Station is connected via a different NetworkConnectionProfile.

Figure 19. Sequence Diagram: Migrate to new ConnectionProfile

8	**Error Handling**	n/a
9	**Remarks**	As in line with B12 - Reset - With Ongoing Transaction, when there are ongoing transactions, the Charging Station waits for these to be finished before performing the Reset and then connecting to a different CSMS. When an operator wants to perform an immediate switch, he should stop the transactions first.

B10 - Migrate to new NetworkConnectionProfile - Requirements

Table 54. B10 - Requirements

ID	Precondition	Requirement definition	Note
B10.FR.01	On receipt of a SetVariablesRequest, containing Configuration Variable `NetworkConfigurationPriority` AND the NetworkProfile slots in the message all contain valid configurations	The Charging Station SHALL send SetVariablesResponse with status *Accepted*, or *RebootRequired*.	
B10.FR.02	On receipt of a SetVariablesRequest, containing Configuration Variable `NetworkConfigurationPriority` AND any of the NetworkProfile slots in the message does not contain a valid configuration	The Charging Station SHALL send SetVariablesResponse with status *Rejected*.	The optional element *statusInfo* can be used to provide more information.
B10.FR.03	When connecting fails	The Charging Station SHALL make the number of attempts as configured in `NetworkProfileConnectionAttempts` per entry of `NetworkConfigurationPriority`.	If after the number of attempts the connection fails, the Charging Station should go back to the old NetworkConnectionProfile.
B10.FR.04	After a reboot	The Charging Station SHALL begin connecting to the first entry of `NetworkConfigurationPriority`	
B10.FR.05		It is RECOMMENDED to set the Charging Station to Inoperative (via ChangeAvailabilityRequest) to ensure that no new transactions can be started and wait until the transaction message queue in the Charging Station is empty before sending the ResetRequest. Otherwise the Charging Station might send transaction related messages to the new CSMS that has not received the start of the Transaction, and the old system will miss the ended messages. To determine if there are still transaction for an ongoing transaction in the queue, the getTransactionStatusRequest message can be used.	
B10.FR.06		The Charging Station SHALL disconnect from the old CSMS, before trying to connect to the new CSMS.	

2.3. Resetting a Charging Station

B11 - Reset - Without Ongoing Transaction

Table 55. B11 - Reset - Without Ongoing Transaction

No.	Type	Description
1	**Name**	Reset - Without Ongoing Transaction
2	**ID**	B11
	Functional block	B. Provisioning
3	**Objective(s)**	To enable the CSMS to request a Charging Station to reset itself or an EVSE, while there is no ongoing transaction.
4	**Description**	This use case covers how the CSMS can request the Charging Station to reset itself or an EVSE by sending ResetRequest. (If ResetRequest contains an optional paramater *evseId*, then only a reset of the specific EVSE is requested.) This could for example be necessary if the Charging Station is not functioning correctly.
	Actors	Charging Station, CSMS, CSO

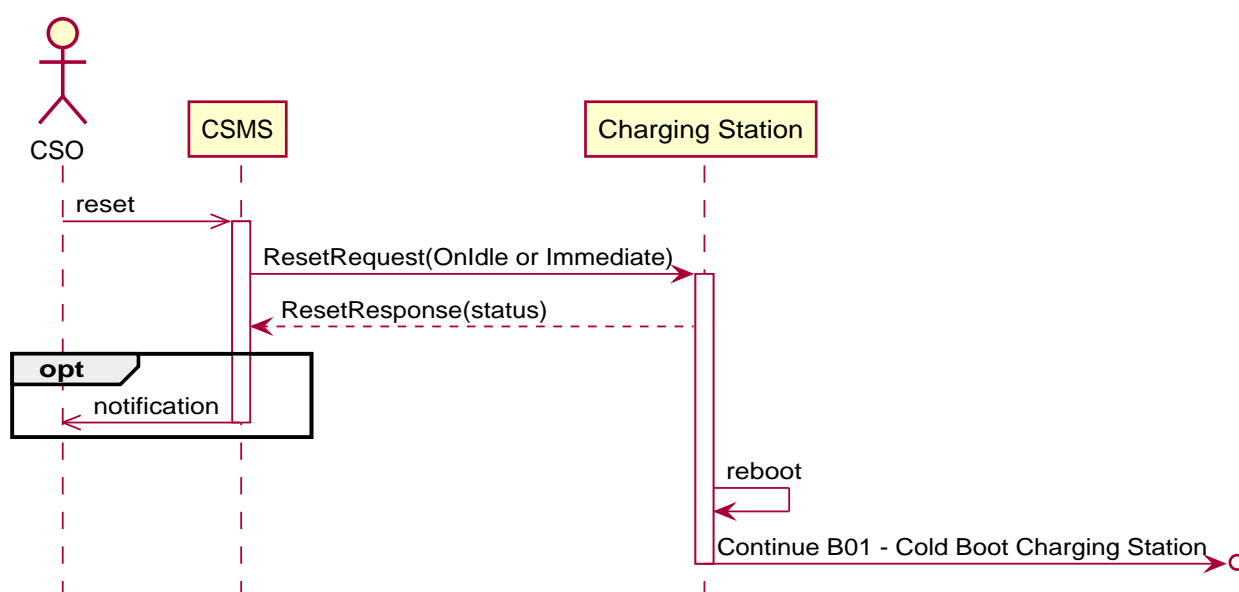| No.	Type	Description
	Scenario description	**1.** The CSO requests the CSMS to reset the Charging Station or EVSE. **2.** The CSMS sends ResetRequest requesting the Charging Station to reset itself or EVSE. **3.** The CSMS requests for an OnIdle or Immediate reset. **4.** The Charging Station responds with ResetResponse, indicating whether the Charging Station is able to reset itself or EVSE. **5.** The CSMS sends an optional notification to the CSO. **6.** Only if no evseId was supplied, then after the reset, the Charging Station will proceed as in use case B01.
	Alternative scenario(s)	B12 - Reset With Ongoing Transaction
5	**Prerequisite(s)**	No transaction is ongoing.
6	**Postcondition(s)**	**Successful postcondition:** The Charging Station was able to reset itself or EVSE. **Failure postcondition:** The Charging Station *not* was able to reset itself or EVSE.

Figure 20. Sequence Diagram: Reset Without Transaction

7	**Error handling**	n.a
8	**Remark(s)**	Persistent states: for example, EVSE set to *Unavailable* SHALL persist. The Charging Station responds with ResetResponse.

B11 - Reset - Without Ongoing Transaction - Requirements

Table 56. B11 - Requirements

ID	Precondition	Requirement definition
B11.FR.01	When the Charging Station receives a ResetRequest.	The Charging Station SHALL respond with a ResetResponse.
B11.FR.02	If the status was set to *Inoperative* by the CSMS.	After a reboot of the Charging Station, the EVSEs SHALL return to the state *Unavailable* as prior to the reboot.
B11.FR.03	B11.FR.01 AND no *evseId* parameter is supplied AND ResetResponse was *Accepted*.	The Charging Station SHALL start a reboot.
B11.FR.04	B11.FR.03	The Charging Station SHALL proceed as described in use case B01 - Cold Boot Charging Station.

ID	Precondition	Requirement definition
B11.FR.05	If the status of an EVSE was *Reserved*.	After a reboot of the Charging Station or EVSE, the EVSE(s) SHALL return to the state *Reserved*.
B11.FR.06	B11.FR.01 AND For example there is a firmware update ongoing that cannot be interrupted.	The Charging Station SHALL respond with a status *Rejected*.
B11.FR.07	B11.FR.01 AND Charging Station cannot perform the reset now, but has scheduled the reset for later	The Charging Station SHALL respond with a status *Scheduled*.
B11.FR.08	B11.FR.01 AND an *evseId* parameter is supplied AND ResetResponse was *Accepted*.	The Charging Station SHALL start a reboot of EVSE that is referred to by *evseId* parameter.
B11.FR.09	B11.FR.01 AND an *evseId* parameter is supplied AND Charging Station does not support resetting an individual EVSE	The Charging Station SHALL return a ResetResponse *Rejected*
B11.FR.10	When the Charging Station supports resetting of an individual EVSE	The Charging Station SHOULD set the device model variable AllowReset to true for the EVSE.

B12 - Reset - With Ongoing Transaction

Table 57. B12 - Reset - With Ongoing Transaction

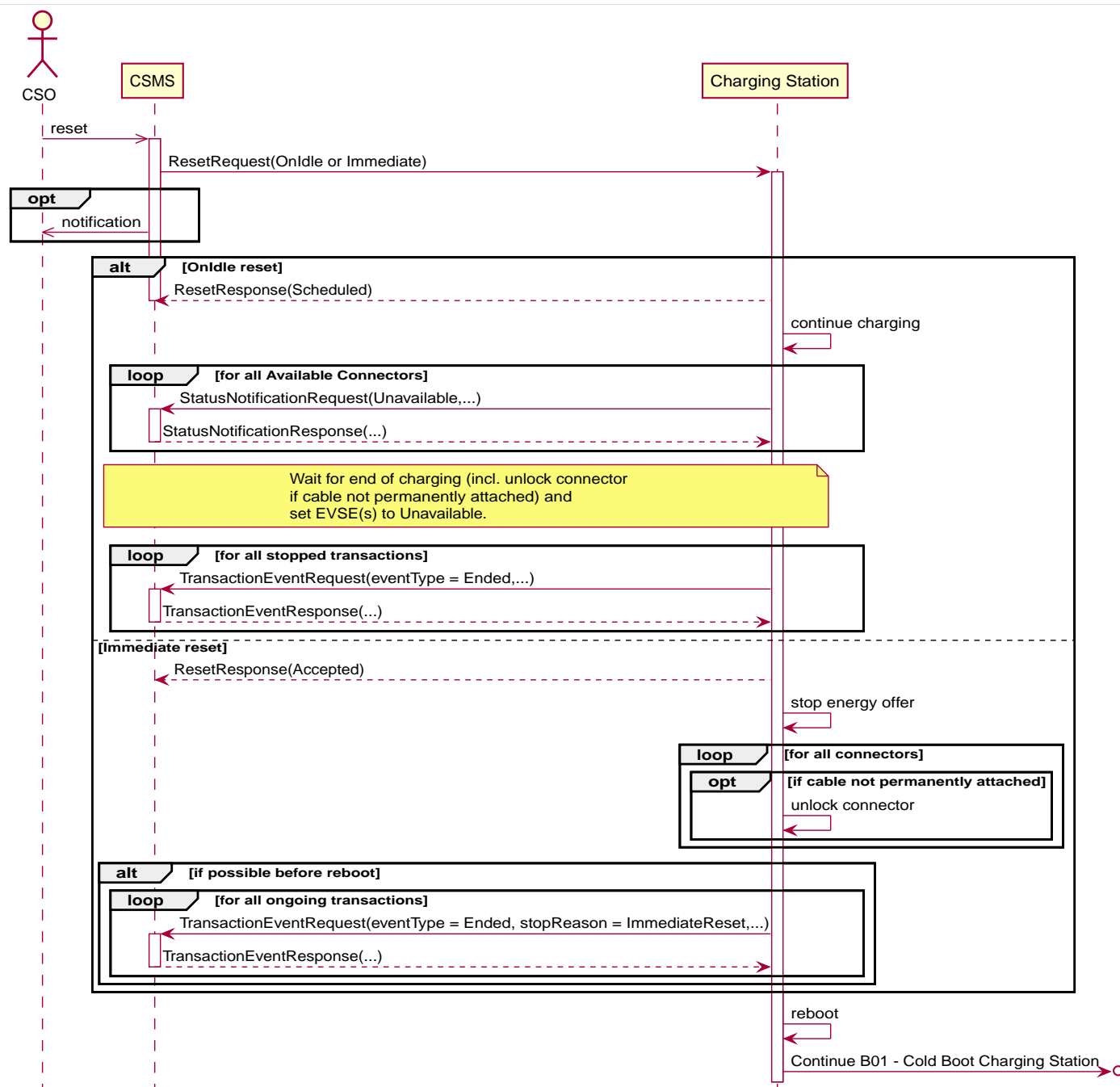| No.	Type	Description
1	**Name**	Reset - With Ongoing Transaction
2	**ID**	B12
	Functional block	B. Provisioning
3	**Objective(s)**	To enable the CSMS to request a Charging Station to reset itself or EVSE, while there is an ongoing transaction.
4	**Description**	This use case covers how the CSMS can request the Charging Station to reset itself or an EVSE by sending ResetRequest. (If ResetRequest contains an optional paramater *evseId*, then only a reset of the specific EVSE is requested.) This could for example be necessary if the Charging Station is not functioning correctly. The CSMS has the possibility to let the Charging Station end all transactions itself and reboot or wait until all ongoing transactions are ended normally (by an EV user) and then reboot.
	Actors	Charging Station, CSMS, CSO
	Scenario description	**1.** The CSO requests the CSMS to reset the Charging Station or EVSE. **2.** The CSMS sends ResetRequest requesting the Charging Station to reset itself or EVSE. **3a.** On receipt of an OnIdle reset, the Charging Station responds with ResetResponse(Scheduled), indicating the Charging Station will try to reset itself or EVSE after all ongoing transactions have ended. The Charging Station continues charging and sets all EVSEs (or only the one provided in the request, if *evseId* was supplied) that are Available to status *Unavailable*, waits until all transactions are finished and all TransactionEventRequest (eventType = Ended) messages are sent. **3b.** On receipt of an Immediate reset, the Charging Station responds with ResetResponse(Accepted), indicating the Charging Station will try to reset itself or EVSE. The Charging Station attempts to terminate any transaction (or only those running on the EVSE provided in the request, if *evseId* was supplied) in progress, and sending a TransactionEventRequest (eventType = Ended) message. **4.** Only if no evseId was supplied the Charging Station reboots and returns to a state as just having been booted, B01 - Cold Boot Charging Station applies.
	Alternative scenario(s)	B11 - Reset Without Ongoing Transaction
5	**Prerequisite(s)**	A transaction is ongoing.
6	**Postcondition(s)**	**Successful postcondition:** The Charging Station was able to reset itself or EVSE. **Failure postcondition:** The Charging Station *not* was able to reset itself or EVSE.

Figure 21. Sequence Diagram: Reset With Ongoing Transaction

7	Error handling	After having accepted the ResetRequest, TransactionEventRequest messages that cannot be delivered to the CSMS MUST be queued.
8	Remark(s)	n/a

B12 - Reset - With Ongoing Transaction - Requirements

Table 58. B12 - Requirements

ID	Precondition	Requirement definition
B12.FR.01	When the Charging Station receives a ResetRequest(OnIdle)	The Charging Station SHALL respond with a ResetResponse(Scheduled), to indicate whether the Charging Station will attempt to reset itself or EVSE after all transactions on Charging Station or EVSE have ended.
B12.FR.02	When the Charging Station receives a ResetRequest(Immediate)	The Charging Station SHALL respond with a ResetResponse(Accepted), to indicate whether the Charging Station will attempt to reset itself or EVSE.

ID	Precondition	Requirement definition
B12.FR.03	If no *evseId* is supplied AND If any transaction is in progress and an OnIdle reset is received.	The transaction of the Charging Station SHALL be terminated normally, before the reboot, as in E06 - Stop Transaction.
B12.FR.04	If no *evseId* is supplied AND If any transaction is in progress and an Immediate Reset is received.	The Charging Station SHALL attempt to terminate any transaction in progress and send a TransactionEventRequest (eventType = Ended) message before performing a reboot.
B12.FR.05	If an Immediate Reset is received and the TransactionEventResponse is not received within timeout.	The Charging Station SHALL queue the TransactionEventRequest, reboot and resend the TransactionEventRequest after the reboot.
B12.FR.06	If the status was set to *Inoperative* by the CSMS.	After a reboot of the Charging Station or EVSE, the EVSE(s) SHALL return to the state *Unavailable* as prior to the reboot.
B12.FR.07	If an *evseId* is supplied AND If a transaction is in progress on the EVSE and an OnIdle reset is received.	The transaction on the EVSE SHALL be terminated normally, before the reboot, as in E06 - Stop Transaction.
B12.FR.08	If an *evseId* is supplied AND If a transaction is in progress on the EVSE and an Immediate Reset is received.	The Charging Station SHALL attempt to terminate the transaction in progress on the EVSE and send a TransactionEventRequest (eventType = Ended) message before performing a reboot.
B12.FR.09	B12.FR.01 AND an *evseId* parameter is supplied AND Charging Station does not support resetting an individual EVSE	The Charging Station SHALL return a ResetResponse *Rejected*