

Open Charge Point Protocol 1.6

Table of Contents

- 1. Scope 6
- 2. Terminology and Conventions 7
 - 2.1. Conventions 7
 - 2.2. Definitions 7
 - 2.3. Abbreviations 8
 - 2.4. References 9
- 3. Introduction 10
 - 3.1. Document structure..... 10
 - 3.2. Feature Profiles..... 10
 - 3.3. General views of operation 13
 - 3.4. Local Authorization & Offline Behavior 15
 - 3.5. Transaction in relation to Energy Transfer Period 18
 - 3.6. Transaction-related messages 20
 - 3.7. Connector numbering 21
 - 3.8. ID Tokens 21
 - 3.9. Parent idTag..... 22
 - 3.10. Reservations 22
 - 3.11. Vendor-specific data transfer 23
 - 3.12. Smart Charging 23
 - 3.13. Time zones 33
- 4. Operations Initiated by Charge Point 34
 - 4.1. Authorize 34
 - 4.2. Boot Notification..... 34
 - 4.3. Data Transfer..... 35
 - 4.4. Diagnostics Status Notification 36
 - 4.5. Firmware Status Notification 37
 - 4.6. Heartbeat 37
 - 4.7. Meter Values 38
 - 4.8. Start Transaction 39
 - 4.9. Status Notification 40
 - 4.10. Stop Transaction..... 45
- 5. Operations Initiated by Central System 48
 - 5.1. Cancel Reservation..... 48
 - 5.2. Change Availability 48
 - 5.3. Change Configuration 49
 - 5.4. Clear Cache 49

5.5. Clear Charging Profile	50
5.6. Data Transfer.....	50
5.7. Get Composite Schedule	50
5.8. Get Configuration	51
5.9. Get Diagnostics	51
5.10. Get Local List Version	52
5.11. Remote Start Transaction.....	52
5.12. Remote Stop Transaction	53
5.13. Reserve Now	54
5.14. Reset	55
5.15. Send Local List	56
5.16. Set Charging Profile	56
5.17. Trigger Message	59
5.18. Unlock Connector.....	60
5.19. Update Firmware	60
6. Messages.....	62
6.1. Authorize.req	62
6.2. Authorize.conf.....	62
6.3. BootNotification.req	62
6.4. BootNotification.conf.....	63
6.5. CancelReservation.req	64
6.6. CancelReservation.conf.....	64
6.7. ChangeAvailability.req	65
6.8. ChangeAvailability.conf	65
6.9. ChangeConfiguration.req	65
6.10. ChangeConfiguration.conf	66
6.11. ClearCache.req	66
6.12. ClearCache.conf	66
6.13. ClearChargingProfile.req	67
6.14. ClearChargingProfile.conf	67
6.15. DataTransfer.req	68
6.16. DataTransfer.conf	68
6.17. DiagnosticsStatusNotification.req	68
6.18. DiagnosticsStatusNotification.conf	69
6.19. FirmwareStatusNotification.req.....	69
6.20. FirmwareStatusNotification.conf.....	69
6.21. GetCompositeSchedule.req	69
6.22. GetCompositeSchedule.conf	70

6.23. GetConfiguration.req	71
6.24. GetConfiguration.conf	71
6.25. GetDiagnostics.req	71
6.26. GetDiagnostics.conf	72
6.27. GetLocalListVersion.req	72
6.28. GetLocalListVersion.conf	73
6.29. Heartbeat.req	73
6.30. Heartbeat.conf	73
6.31. MeterValues.req	73
6.32. MeterValues.conf	74
6.33. RemoteStartTransaction.req	74
6.34. RemoteStartTransaction.conf	74
6.35. RemoteStopTransaction.req	75
6.36. RemoteStopTransaction.conf	75
6.37. ReserveNow.req	75
6.38. ReserveNow.conf	76
6.39. Reset.req	76
6.40. Reset.conf	76
6.41. SendLocalList.req	77
6.42. SendLocalList.conf	78
6.43. SetChargingProfile.req	78
6.44. SetChargingProfile.conf	78
6.45. StartTransaction.req	79
6.46. StartTransaction.conf	80
6.47. StatusNotification.req	80
6.48. StatusNotification.conf	81
6.49. StopTransaction.req	81
6.50. StopTransaction.conf	82
6.51. TriggerMessage.req	82
6.52. TriggerMessage.conf	83
6.53. UnlockConnector.req	83
6.54. UnlockConnector.conf	83
6.55. UpdateFirmware.req	83
6.56. UpdateFirmware.conf	84
7. Types	85
7.1. AuthorizationData	85
7.2. AuthorizationStatus	85
7.3. AvailabilityStatus	86

7.4. AvailabilityType	86
7.5. CancelReservationStatus	86
7.6. ChargePointErrorCode	87
7.7. ChargePointStatus	87
7.8. ChargingProfile	89
7.9. ChargingProfileKindType	91
7.10. ChargingProfilePurposeType	91
7.11. ChargingProfileStatus	92
7.12. ChargingRateUnitType	92
7.13. ChargingSchedule	92
7.14. ChargingSchedulePeriod	93
7.15. CiString20Type	94
7.16. CiString25Type	94
7.17. CiString50Type	94
7.18. CiString255Type	95
7.19. CiString500Type	95
7.20. ClearCacheStatus	95
7.21. ClearChargingProfileStatus	95
7.22. ConfigurationStatus	96
7.23. DataTransferStatus	96
7.24. DiagnosticsStatus	96
7.25. FirmwareStatus	97
7.26. GetCompositeScheduleStatus	97
7.27. IdTagInfo	98
7.28. IdToken	98
7.29. KeyValue	98
7.30. Location	99
7.31. Measurand	99
7.32. MessageTrigger	100
7.33. MeterValue	101
7.34. Phase	101
7.35. ReadingContext	102
7.36. Reason	102
7.37. RecurrencyKindType	103
7.38. RegistrationStatus	103
7.39. RemoteStartStopStatus	103
7.40. ReservationStatus	104
7.41. ResetStatus	104

7.42. ResetType	104
7.43. SampledValue	105
7.44. TriggerMessageStatus	106
7.45. UnitOfMeasure	106
7.46. UnlockStatus	107
7.47. UpdateStatus	107
7.48. UpdateType	108
7.49. ValueFormat	108
8. Firmware and Diagnostics File Transfer	109
8.1. Download Firmware	109
8.2. Upload Diagnostics.....	109
9. Standard Configuration Key Names & Values	110
9.1. Core Profile	110
9.2. Local Auth List Management Profile	119
9.3. Reservation Profile	120
9.4. Smart Charging Profile	120
Appendix A: New in OCPP 1.6	122
A.1. Updated/New Messages:.....	122

Interface description between Charge Point and Central System

Document Version	1.6
Document Status	FINAL
Document Release Date	2015-10-08

Copyright © 2010 – 2015 Open Charge Alliance. All rights reserved.

This document is made available under the **Creative Commons Attribution-*

Version History

Version	Date	Author	Description
1.6	2015-10-08	Robert de Leeuw <i>IHomer</i> Reinier Lamers <i>The New Motion</i> Brendan McMahon <i>ESB ecars</i> Lambert Muhlenberg <i>Alfen</i> Patrick Rademakers <i>IHomer</i> Sergiu Tcaciuc <i>smartlab</i> Klaas van Zuuren <i>ElaadNL</i>	1.6 Final Release. For changes relative to 1.5, see appendix New in OCPP 1.6 .

Version	Date	Author	Description
1.5	2012-06-01	Franc Buve	<p>Specification ready for release. Includes:</p> <p>CR-01 Authentication/authorization lists</p> <p>CR-02 Interval meter readings</p> <p>CR-03 Charge point reservation</p> <p>CR-04 Generic data transfer</p> <p>CR-05 More detailed status notifications</p> <p>CR-06 Query configuration parameters</p> <p>CR-07 Timestamp in BootNotification mandatory</p> <p>CR-08 Response to StartTransaction.req with status other than Accepted is not clearly defined</p> <p>CR-09 Increase size of firmwareVersion in BootNotification</p>
1.2	2011-02-21	Franc Buve	

Version	Date	Author	Description
1.0	2010-10-19	Franc Buve	Final version approved by e-laad.nl. Identical to version 0.12.

1. Scope

本文档定义了充电点和中央系统之间使用的协议。如果协议需要一方或另一方的某种行动或响应，那么这将在本文件中说明。

规范未定义通信技术。任何技术都行，只要支持TCP/IP连接。 6.06V ----1.51V 4.035 ---1.007

2. Terminology and Conventions

2.1. Conventions

本文件中的关键词“必须”、“必须不”、“要求”、“应”、“不应”、“应”、“应”、“不应”、“建议”、“可能”和“备选”应按照[RFC2119]中的说明加以解释，但须遵守以下补充澄清条款：

“在特定情况下的有效理由”一语涉及使用“应当”、“应当不应当”、“应当建议”和“不应当建议”等术语，应被视为是指技术上有效的理由，例如缺乏必要的硬件来支持收费点设计的功能：为本规范的目的，它具体排除了基于商业或其他非技术原因作出的决定，例如执行费用或使用可能性。

除“范围”和“术语和公约”外，所有章节和附录都是规范性的，除非明确指出它们具有信息性。

2.2. 定义

本节包含本文档中使用的术语。

中央系统	收费点管理系统：管理收费点并拥有授权用户使用其收费点的信息的中央系统.
CiString	不区分大小写的字符串。只允许可打印的ASCII.
充电点	充电点是电动汽车可以充电的物理系统。充电点有一个或多个连接器.
充电配置文件	通用充电配置文件，用于不同类型的配置文件。包含有关配置文件的信息，并保存充电时间表。在OCPP的未来版本中，它可能持有超过1个充电时间表
收费时间表	充电配置文件的一部分。定义一个充电电源或电流限制块。可以包含开始时间和长度
收费会议	允许EV请求能量的事务的一部分
综合充电时间表	充电点计算的充电时间表。这是计算所有活动时间表和可能的局部限制在收费点的结果。此外，IEC15118的限制也可以考虑在内.

连接器	本规范中使用的术语“连接器”是指充电点上独立操作和管理的电源插座。这通常对应于单个物理连接器，但在某些情况下，单个出口可能具有多个物理插座类型和/或栓接电缆/连接器安排，以方便不同的车辆类型(例如。四轮电动汽车和电动滑板车)。
控制引导信号	充电点用于通知EV最大充电功率或电流限制的信号，如[IEC61851-1]所定义的]。
能源转换期	电动汽车选择获取所提供的能量或返回能量的时间。多个能量转移周期是可能的交易。
本地控制器	智能充电基础设施中的可选设备。位于酒店内，有多个充电点与之相连。位于收费点和中央系统之间。理解并讲OCPP消息。通过使用OCPP智能充电消息控制其他充电点的功率或电流。可以是电荷点本身。
OCPP-J	OCPP通过JSON通过Web
OCPP-S	通过SOAP套接字OCPP
相位旋转	定义能量计（或如果没有，网络连接）和充电点连接器之间相位的布线顺序。
交易	收费过程的一部分，从所有相关的先决条件(例如。授权，插入插头)被满足，并在充电点不可撤销地离开此状态的时刻结束。
字符串	案件敏感字符串。只允许可打印的ASCII。消息和枚举中的所有字符串都是区分大小写的，除非另有明确说明。

2.3. Abbreviations

CSL	Comma Separated List 逗号分隔列表
CPO	Charge Point Operator 充电点操作员
DNS	Domain Name System 域名系统
DST	Daylight Saving Time 存储时间
EV	Electrical Vehicle 电动汽车
EVSE	Electric Vehicle Supply Equipment [IEC61851-1] 电动汽车供应设备
FTP(S)	File Transport Protocol (Secure) 文件传输协议（安全）
HTTP(S)	HyperText Transport Protocol (Secure) 超文本传输协议（安全）
ICCID	Integrated Circuit Card Identifier 集成电路卡标识符

IMSI	International Mobile Subscription Identity 国际移动订阅身份
JSON	Java Simple Object Notation Java简单对象标记
NAT	Native Address Translation 本地地址翻译
PDU	Protocol Data Unit 协议数据单元
SC	Smart Charging 智能充电
SOAP	Simple Object Access Protocol 简单对象访问协议
URL	Uniform Resource Locator 统一资源定位器
RST	3 phase power connection, Standard Reference Phasing 三相电源连接, 标准参考相位
RTS	3 phase power connection, Reversed Reference Phasing 三相电源连接, 反向参考相位
SRT	3 phase power connection, Reversed 240 degree rotation 三相电源连接, 反转240度旋转
STR	3 phase power connection, Standard 120 degree rotation 三相电源连接, 标准120度旋转
TRS	3 phase power connection, Standard 240 degree rotation 三相电源连接, 标准240度旋转
TSR	3 phase power connection, Reversed 120 degree rotation 三相电源连接, 反转120度旋转
UTC	Coordinated Universal Time 协调世界时间

2.4. References

[IEC61851-1]	
[OCPP1.5]	
[OCPP_1.6CT]	
[OCPP_IMP_J]	“OCPP JSON Specification”
[OCPP_IMP_S]	“OCPP SOAP
[RFC2119]	“Key March

3. 引言

这是OCPP版本1.6的规范。

OCPP是一种标准的开放协议，用于充电点和中央系统之间的通信，旨在适应任何类型的充电技术。

OCPP1.6引入了适应市场的新功能：智能充电、在Websockets上使用JSON的OCPP、更好的诊断可能性（原因）、更多的充电点状态和触发消息。OCPP1.6基于OCPP1.5，对所有已知的歧义进行了一些新的特性和大量的文本改进、澄清和修复。由于改进和新特性，OCPP1.6与OCPP1.5不兼容。。

有关更改的完整列表，请参见：OCPP1.6中的New。。

本介绍性章节将在以下章节中解释一些基本概念。章节：由收费点发起的操作和由中央系统发起的操作描述了协议支持的操作。确切的消息及其参数详见章节：消息和数据类型在章节：类型中描述。定义的配置键在章节中描述：标准配置键名和值。

3.1. 文件结构

随着OCPP1.6的引入，OCPP有两种不同的味道；除了基于SOAP的实现之外，还有可能使用更紧凑的JSON替代方案。为了避免在实现类型的通信中出现混淆，我们建议使用不同的后缀-J和-S来表示JSON或SOAP。在通用术语中，这将是JSON的OCPP-J和SOAP的OCPP-S。。

为了支持不同的口味，OCPP标准分为多个文档。基础文档（您现在正在阅读的文档）包含技术协议规范。技术协议规范必须与传输协议规范之一一起使用。OCPP SOAP规范包含实现OCPP-S实现所需的实现规范。对于OCPP-J，必须使用OCPP JSON规范。。

为了改进中央系统和收费点之间的互操作性，建议满足OCPP1.6符合性测试文档中所述的要求。

3.2. 专题简介

本节内容丰富。

在OCPP1.6中，特征和相关消息被分组在配置文件中。根据所需的功能，实现人员可以选择实现以下一个或多个配置文件。

项目名称	说明
核心	基本充电点功能与OCPP1.5[OCPP1.5]相当，不支持固件更新、本地授权列表管理和保留。
固件管理	支持固件更新管理和诊断日志文件下载。
本地自动列表管理	功能来管理收费点中的本地授权列表。
保留	支持预订收费点。
智能充电	支持基本智能充电，例如使用控制试点。
远程触发	支持远程触发充电点发起的消息

客户可以使用这些配置文件来确定OCPP1.6产品是否具有业务案例所需的功能。如果产品符合OCPP1.6规范，则符合性测试将测试每个配置文件。

需要实现核心配置文件。其他配置文件是可选的。

当配置文件核心、固件管理、本地Auth列表管理和保留实现时，将涵盖来自OCPP1.5[OCPP1.5]的所有功能。

所有消息在其配置文件中的分组可以在下表中找到。

信息	核心	固件管理	本地自动列表管理	保留	智能充电	远程触发
Authorize 授权	X					
BootNotification 启动通知	X					
ChangeAvailability 更改可用性	X					
ChangeConfiguration 更改配置	X					
ClearCache 清空缓存	X					
DataTransfer 数据传输	X					

Message	Core	Firmware Management	Local Auth List Management	Reservation	Smart Charging	Remote Trigger
GetConfiguration 获取配置	X					
Heartbeat 心跳	X					
MeterValues 仪表值	X					
RemoteStart Transaction 远程启动事务	X					
RemoteStopTransaction 远程停止事务	X					
Reset 复位	X					
StartTransaction 开始传输	X					
StatusNotification 状态通知	X					
StopTransaction 停止传输	X					
UnlockConnector 解锁连接器	X					
GetDiagnostics 获取诊断		X				
DiagnosticsStatusNotification 诊断STATUS通知		X				
FirmwareStatusNotification 固件状态通知		X				
UpdateFirmware 更新固件		X				
GetLocalListVersion 获取本地列表版本			X			
SendLocalList 发送本地列表			X			
CancelReservation 取消预约				X		

Message	Core	Firmware Managemen t	Local Auth List Managemen t	Reservation	Smart Charging	Remote Trigger
ReserveNow	立刻准备			X		
ClearChargin gProfile	清除的充电配置文件				X	
GetComposit eSchedule	得到综合时间表				X	
SetChargingP Rofile	设置充电配置文件				X	
TriggerMess Age	触发信息					X

特定充电配置文件的支持由支持的特征配置文件配置键报告.

3.3. General views of operation 操作的一般看法

本节内容丰富.

下图描述了两两种情况下收费点与中央系统之间业务的一般观点:

- 1. 要求验证卡和发送收费交易状态的收费点,
- 2. 中央系统要求充电点更新其固件。

下图中的箭头标签表示在调用操作期间交换的PDU。 这些PDU在消息部分中详细定义.

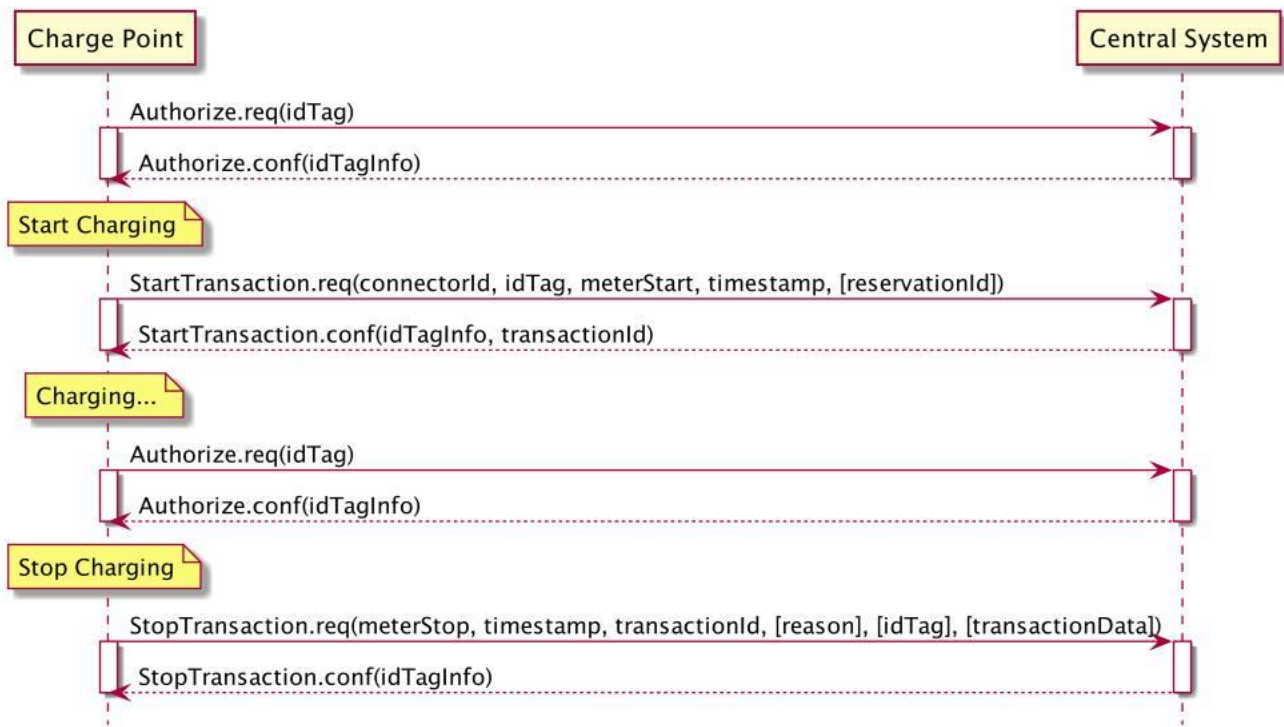


图1。顺序图：启动和停止事务的示例

当充电点需要为电动汽车充电时，它需要首先对用户进行身份验证，然后才能启动充电。如果用户被授权，充电点通知中央系统，它已经开始充电。

当用户希望从充电点拔出电动汽车时，充电点需要验证用户要么是发起充电的人，要么是用户在同一组中，从而允许终止充电。一旦授权，充电点通知中央系统充电已停止。

NOTE

如果所呈现的idTag与为启动事务而呈现的idTag相同，则在停止事务之前，收费点不得发送 Authorize.req。

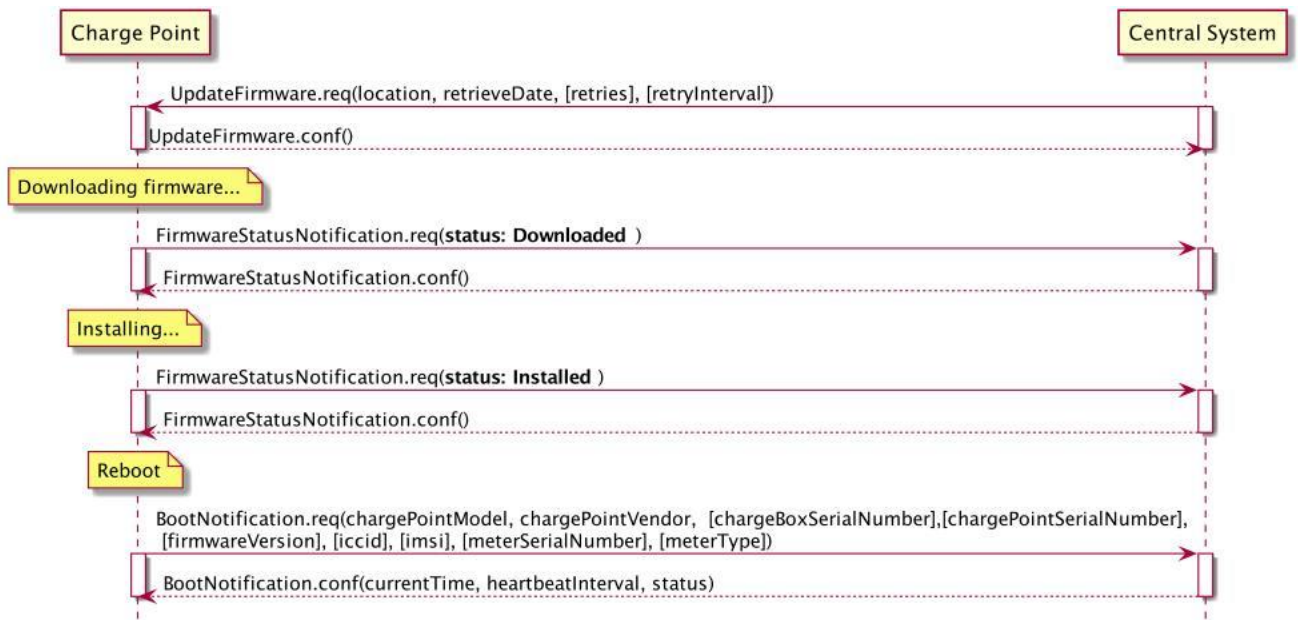


图2。序列图：固件更新的示例

当充电点需要用新固件更新时，中央系统通知充电点可以开始下载新固件的时间。充电点应通知中央系统后，每一步，因为它下载和安装新的固件。

3.4. 本地授权和离线行为

本节是规范性的。

如果没有通信，甚至没有中央系统，收费点的设计是独立运作的。在这种情况下，收费点据说是离线的。

为了提高用户的体验，收费点可能支持标识符的本地授权，使用授权缓存和/或本地授权列表。

这允许(A)离线时用户的授权，(B)当充电点和中央系统之间的通信缓慢时，更快（明显）的授权响应时间。

本地授权脱机配置密钥控制充电点在使用授权缓存和/或本地授权列表脱机时是否授权用户。。

本地预授权配置密钥控制收费点是否将使用授权缓存和/或本地授权列表启动事务，而无需等待中央系统的授权响应。。

收费点可能支持（自动）授权任何呈现的标识符时，离线，以避免拒绝收费的真正用户，不能明确授权的本地授权列表/授权缓存条目。此功能将在未知离线授权中更详细地解释。

3.4.1. Authorization Cache 授权缓存

充电点可能实现一个授权缓存，该缓存自动维护由中央系统成功授权的先前呈现的标识符的记录。（成功含义：在包含id标记的消息上收到的响应）

如果实现，授权缓存应该符合以下语义：

- 缓存包含所有最新接收到的标识符(即。有效和无效)。
- 使用所有接收到的Id标记信息(从Authorize.conf、Start Transaction.conf和Stop Transaction.conf)更新缓存)
- 当缓存条目的有效性过期时，将在缓存中更改为过期。
- 当接收到用于缓存中的标识符的ID标记信息时，它将被更新。
- 如果接收到新的标识符授权数据，并且授权缓存已满，则收费点将删除任何未验证的条目，然后在必要时删除最古老的有效条目，为新条目腾出空间。
- 缓存值应该存储在非易失性内存中，并且应该在重新启动和断电时持久化。
- 当一个标识符以NOT-Valid形式存储在缓存中，并且电荷点在线时：一个Authorize.req应该被发送到中央系统以检查标识符的当前状态。

授权缓存的操作，当存在时，由授权缓存启用配置密钥报告(并在可能的情况下控制)。

3.4.2. Local Authorization List 本地授权名单

本地授权列表是可以与中央系统同步的标识符列表。

列表包含所有（或选择）标识符的授权状态和授权状态/过期日期。

本地授权列表中的标识符可以标记为有效、过期、（暂时）阻塞或黑名单，分别对应于接受/当前TX、过期、阻塞和无效的Id标记信息状态值。

这些值可用于向用户提供更细粒度的信息(例如。通过显示消息)在本地授权期间。

本地授权列表应该由非易失性内存中的充电点维护，并且应该在重新启动和断电时持久化。

支持本地授权列表的收费点应该实现配置键：本地自动列表最大长度，这使中央系统有一种方法来知道收费点中本地授权列表元素的最大可能数量

收费点指示本地授权列表是否由支持的特征配置密钥的值中是否存在本地Auth列表管理元素支持。是否启用本地授权列表由本地AuthList启用配置密钥报告和控制。

中央系统可以通过以下方式同步此列表：(1)发送完整的标识符列表以替换本地授权列表；(2)发送更改列表(添加、更新、删除)以应用于本地授权列表。支持此操作的操作是获取本地列表版本并发送本地列表。

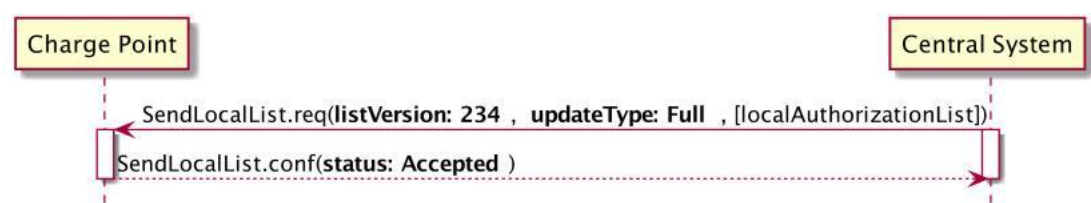


Figure 3. Sequence Diagram: Example of a full local authorization list update

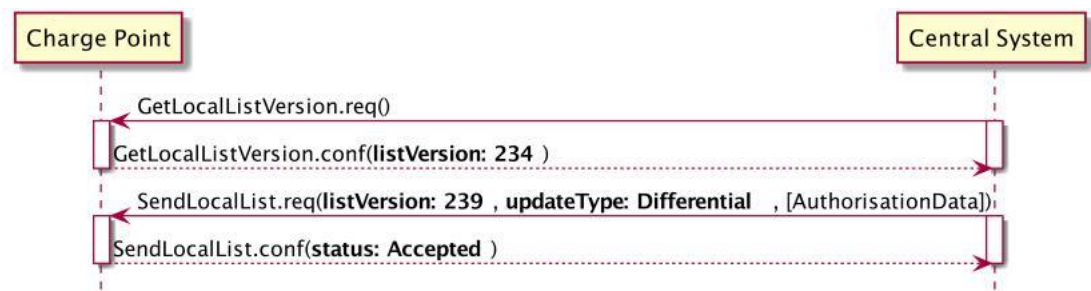


图4。序列图：差分本地授权列表更新的示例

充电点不应以任何其他方式修改授权列表的内容，除非从中央系统收到发送本地列表PDU。

NOTE

本地授权列表与例如Start Transaction.conf消息中报告的有效性之间可能发生冲突。当这种情况发生时，充电点将通过发送连接器Id设置为0的状态通知来通知中央系统，错误代码设置为“本地列表冲突”。

3.4.3. Relation between Authorization Cache and Local Authorization List
授权缓存与本地授权列表之间的关系

授权缓存和本地授权列表是不同的逻辑数据结构。本地授权列表中已知的标识符不应添加到授权缓存中。

如果同时支持授权缓存和本地授权列表，则收费点应将本地授权列表条目视为对相同标识符的授权缓存条目具有优先级。

3.4.4. Unknown Offline Authorization 未知离线授权

当脱机时，充电点可能允许自动授权任何“未知”标识符，这些标识符不能由本地授权列表或授权缓存条目显式授权。本地授权列表中存在的标识符，其状态除“接受”（无效、阻塞、过期）外，必须被拒绝。由于时间推移而有效但显然过期的标识符也必须被拒绝。

支持时，允许离线TX对未知ID配置密钥报告（并在可能的情况下控制）未知离线授权功能的操作。

当连接中央服务器被恢复时，收费点将根据与事务相关的消息处理的要求，授权脱机的事务发送启动事务请求。当Start Transaction.conf中的授权状态未被接受，并且事务仍在进行时，收费点将被保留：

- 当无效ID上的停止事务设置为true时：按停止事务中的说明正常停止事务。停止事务请求中的原因字段应设置为DeAuthorized。如果充电点有可能锁定充电电缆，它应该保持充电电缆锁定，直到业主提出他的标识符。。
- 当无效ID上的停止事务设置为false时：只停止向车辆传递能量。

NOTE	在无效标识符的情况下，操作员可能会选择用最小的能量向EV充电，这样EV就可以开走了。此数量由可选配置键控制：无效Id上的最大能量
------	--

3.5. Transaction in relation to Energy Transfer Period 与能源转移期有关的交易

本节内容丰富。

能量转移期是在EV和EVSE之间传递能量的一段时间。交易期间可能有多个能量转移期。

多个能量转移周期可以用任何一种方式分开：（EVSE：电动车充电设备）（EV：电动汽车）

- 一个EVSE启动转移期间，EVSE不提供能量转移的支持
- 由EV引起的转移暂停，在此期间EV仍然与EVSE保持电连接
- EV引起的转移暂停，在此期间EV不与EVSE连接。

中央系统可以从事务期间发送的仪表值推断能量传输周期的开始和结束。

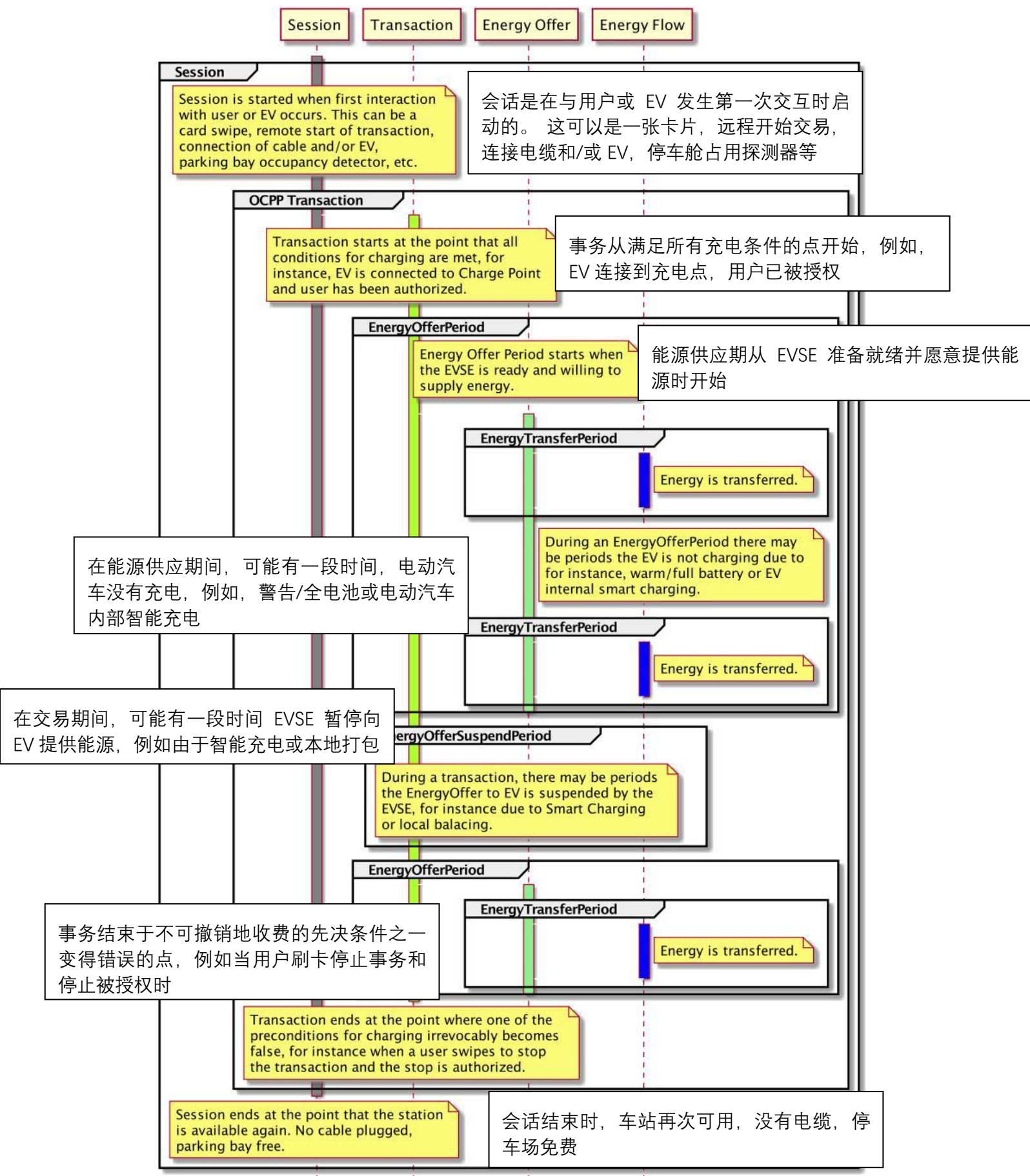


图5。OCPP充电会话和事务定义

3.6. Transaction-related messages 与事务相关的消息

本节是规范性的。

收费点应尽快按时间顺序向中央系统发送与交易有关的信息。与事务相关的消息是Start Transaction.req、Stop Transaction.req和周期性或时钟对齐的Meter Values.req消息。

当脱机时，充电点必须排队任何与交易相关的消息，如果充电点已经联机，它将发送到中央系统。

如果一个收费点有与事务相关的消息排队等待发送到中央系统，那么不与事务相关的新消息可能会立即传递，而无需等待队列被清空。因此，允许在清空与事务相关的消息队列之前发送授权请求或通知请求，这样客户就不会一直等待，紧急通知也不会延迟。

新事务相关消息的传递应该等到队列被清空。这是为了确保与事务相关的消息总是按时间顺序传递。

当中央系统接收到在充电点上排队一段时间的与事务相关的消息时，中央系统将不会意识到这是一个历史消息，除非通过推断，因为各种时间戳在过去是显著的。它应该像任何其他消息一样处理这样的消息。

3.6.1. Error responses to transaction-related messages 事务相关消息的错误响应

当且仅当中央系统反复报告未处理消息时，收费点可以跳过与事务相关的消息。这样的规定是必要的，因为否则，按时间顺序传递每一条与交易有关的消息的要求将导致收费点在软件错误导致中央系统不承认收费点的一条与交易有关的消息之后不能向中央系统传递任何与交易有关的消息。

在OCPP JSON规范和OCPP SOAP规范中定义了什么样的响应或未响应构成了处理消息的失败。

可以使用事务消息尝试和事务消息重试间隔配置密钥来配置收费点重试此类失败的事务相关消息的次数和间隔

当收费点遇到第一次未能传递与事务相关的消息时，它必须再次发送此消息，只要它一直导致无法处理该消息，并且它还没有遇到像其TransactionMessageTurns配置键中指定的那样为该消息处理消息的多个失败。在每次重传之前，它必须等待其事务消息重试间隔键中指定的时间，乘以相同消息的前面传输的次数

例如，考虑一个收费点，它的值是事务消息尝试配置键的“3”，而事务消息重试间隔配置键的值是“60”。它发送停止事务消息，并检测到在中央系统中处理消息的失败。充电点将等待60秒，并重新发送消息。如果出现第二次故障，充电点将等待120秒，然后再发送消息。如果最终尝试失败，充电点应该丢弃消息并继续下一个与事务相关的消息，如果有的话。

3.7. Connector numbering 连接器编号

本节是规范性的。

为了使中央系统能够寻址电荷点的所有连接器，连接器Ids必须始终以相同的方式编号。

连接器编号(连接器IDS)必须如下：

- 第一个连接器的ID必须是1
- 附加连接器必须按顺序编号（不得跳过任何数字）
- 连接器Id数必须永远不高于充电点的连接器总数
- 对于由中央系统满足的操作，连接器ID0被保留用于寻址整个电荷点。
- 对于由充电点发起的操作（报告时），连接器ID0保留给充电点主控制器。

示例：带有3个连接器的充电点：所有连接器必须编号为ID：1、2和3。最好以逻辑的方式对电荷点的连接器进行编号：从左到右，从上到下递增。

3.8. ID Tokens

本节是规范性的。

在大多数情况下，通过本地令牌读取器硬件获取的IDToken数据通常是物理RFID卡的（4或7字节）UID值，通常表示为8/14个十六进制数字字符。

然而，中央系统为远程启动充电而发送到充电点的 idtoken 通常可能是(一次性)虚拟交易授权码，或者故意使用非标准 uid 格式的虚拟 rfid token，以避免可能与真实 uid 值发生冲突。

此外，用作父ID的IDToken数据可能经常使用父ID的共享中心帐户标识符，而不是帐户的第一/主RFID卡的UID。

因此，`idtoken` 类的消息数据元素(包括括号)可以包含对中央系统有意义的任何数据，但受数据类型 (`cistring20type`)的约束(例如，用于确定充电活动的发起者)，而且充电点不能对这些数据的格式或内容做出任何假设(例如，假设它是一个类 `uid`，必须只有十六进制字符和/或一个偶数位)。

NOTE

为了促进互操作性，根据迄今为止在代表物理ISO14443兼容RFID卡UID的IDToken数据的常见实践，建议将这些UID表示为UID字节的十六进制表示。根据ISO14443-3，字节0应该在十六进制字符串中排在第一位

3.9. Parent idTag

这部分是规范的。

中央系统能够将一组身份令牌视为“组”，从而允许组中的任何一个令牌启动事务，并且对于同一个令牌，或者同一个组中的另一个令牌，停止事务。这支持了家庭或企业使用一个或多个共享电动汽车在一个充电合同帐户多司机的共同使用情况。

在 `idtaginfo` 中的可选 `parentid` 元素中指定一个公共组标识符，对令牌(`idtags`)进行分组以达到授权目的: 如果两个 `idtags` 的 `parentid` 标记匹配，则认为它们属于同一组。

NOTE

尽管父ID具有与idTag相同的标称数据类型(`IdToken`)，但此元素的值可能不是`IdToken`的通用格式，并且/或可能不代表实际有效的`IdToken`(例如。它可能是一个共同共享的“帐号”)：因此，父ID值不能用于与呈现的令牌值进行比较(除非它也作为idTag值发生)。

3.10. Reservations

本节内容丰富。

预留一个收费点是可能使用储备现在操作。此操作保留充电点，直到特定idTag的特定到期时间。父idTag可以包含在保留中，以支持“组”保留。可以在充电点上预留特定的连接器，也可以在充电点上预留任何连接器。当在保留连接器（指定时）或在任何连接器(未指定时)上使用保留idTag时，或当达到过期时间时，或当预订被显式取消时，保留将被释放。

3.11. Vendor-specific data transfer

本节内容丰富。

特定于供应商的数据传输机制允许交换数据或消息

在OCPP中标准化。因此，它在OCPP中为实验功能提供了一个框架，可以找到未来OCPP版本的方法。实验可以在不创建新的（可能不兼容的）OCPP方言的情况下进行。其次，它提供了实现特定中央系统和收费点供应商之间商定的额外功能的可能性。

操作供应商特定数据可以由中央系统或收费发起重点。

重点

请非常谨慎地使用，并且只用于可选的功能，因为它将影响您与其他不使用此功能的系统的兼容性选择。我们建议在您的文档中明确提及使用和/或沟通。请考虑咨询开放收费联盟在转到此选项添加功能之前。

3.12. Smart Charging 智能充电

本节是规范性的。

通过智能充电，中央系统获得了影响特定EV的充电功率或电流的能力，或整个充电点/一组充电点的总允许能耗，例如，基于网格连接、腰带上的能量可用性或建筑物的布线。影响电荷功率或电流是基于特定时间点的能量转移极限。这些限制组合在一个充电配置文件。

3.12.1. Charging profile purposes 充电配置文件的目的

充电配置文件由一个充电时间表组成，它基本上是一个时间间隔的列表，其中有它们的最大充电功率或电流，以及一些值来指定时间表的时间周期和重现。

有三种不同类型的充电配置文件，取决于它们的目的：

- *ChargePointMaxProfile* 充电点最大配置文件

在负载平衡场景中，充电点有一个或多个本地充电配置文件，限制充电点的所有连接器共享的功率或电流。中央系统应该配置这样的配置文件与充电配置文件的目的设置为“充电点最大配置文件”。充电点最大配置文件只能设置在充电点连接器ID0。

- *TxDefaultProfile* TX默认配置文件

新事务的默认时间表可能用于强制收费策略。一个例子可以是防止白天收费的政策。对于这一目的的时间表，充电配置文件的目的是应设置为TX默认配置文件。

如果TX默认配置文件设置为连接器ID0，则TX默认配置文件适用于所有连接器。如果Connector Id设置>0，则只适用于该特定连接器。

如果安装了连接器0的TX默认配置文件，并且中央系统发送了一个带有连接器ID>0的新配置文件，则TX默认配置文件将仅用于该特定连接器。

- TX 配置文件

如果存在具有目的TX配置文件的特定事务配置文件，则仅在当前事务的持续时间内，它将用目的TX默认配置文件推翻默认收费配置文件。事务停止后，配置文件将被删除。如果在TX类型的充电配置文件中指定的连接器上没有事务活动，则充电点将丢弃它，并在Set充电配置文件.conf中返回错误状态。

应用于事务的最终进度约束是通过将带有目的电荷点Max配置文件的配置文件与配置文件TX配置文件或TX默认配置文件合并来确定的，以防没有提供目的TX配置文件的配置文件。TX配置文件只应设置在充电点连接器ID>0。.

3.12.2. Stacking charging profiles 堆叠充电剖面

它允许堆叠相同充电配置文件的目的，以描述复杂的日历。例如，一个人可以定义一个充电配置文件的目的TX默认配置文件，持续时间和重复一周，允许在工作日从23: 00h到06: 00h和从00: 00h到24: 00h，并减少电源或电流充电在其他时间。除此之外，还可以定义其他定义此规则异常的TX Default Profiles，例如节假日。

充电配置文件的优先级由它们的堆栈级别参数的值决定。在任何时间点，流行的充电配置文件应该是充电配置文件中具有最高堆栈级别的配置文件，这些配置文件在该时间点有效，由它们的有效从参数和有效到参数决定。

为了避免冲突，不允许在充电点中存在具有相同堆栈级别和目的的多个充电配置文件。每当充电点接收到具有堆栈级别和目的的充电配置文件时，充电点将替换现有配置文件。

注

在更新的充电配置文件（具有相同的堆栈级别和目的）与有效的从日期时间在未来发送，充电点将替换安装的配置文件，并将恢复到默认行为，直到有效从。建议在过去提供一个开始时间，以防止出现空白

3.12.3. Combining charging profile purposes 结合充电配置文件的目的

将指导充电水平的综合时间表是不同充电配置文件用途的当前充电配置文件的组合。

这个综合时间表是通过取每个时间间隔的最小值来计算的。请注意，时间间隔不必是固定长度的，也不必为每个充电配置文件的目的相同。这意味着产生的复合时间表可能包含不同长度的间隔。

在任何时间点，复合时间表中的可用功率或电流，这是合并充电配置文件的时间表，充电点最大配置文件和TX默认配置文件(或TX配置文件)的结果，应小于或等于任何合并时间表中可用功率或电流的最低值。

在充电点配备多个连接器的情况下，充电点最大配置文件的限值是所有连接器组合的限值。所有连接器的组合能量流不应大于充电点最大轮廓设定的极限。

3.12.4. Smart Charging Use Cases 智能充电用例

本节内容丰富。

智能充电可能有许多不同的用途。以下三种典型的智能充电将用于说明智能充电的可能行为：

- 负载平衡
- 中央智能充电
- 本地智能充电

有更复杂的用例可能将上述两个或多个用例组合在一起变成一个更复杂的系统。

Load Balancing 负载平衡

本节内容丰富。

负载平衡用例是关于充电点内部负载平衡的，充电点控制每个连接器的充电时间表。充电点配置了一个固定的限制，例如连接到网格的最大电流。

可选充电计划字段最小充电速率可由充电点用于优化连接器之间的功率分配。该参数通知充电点，低于最小充电率的充电效率低下，从而有可能选择另一种平衡策略。

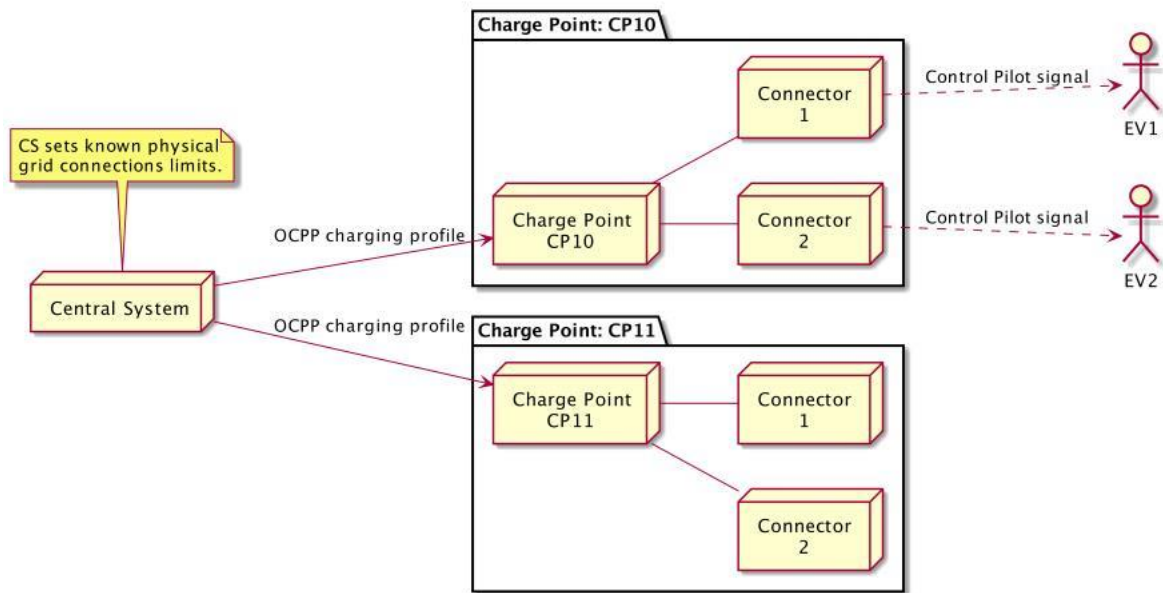


Figure 6. Load balancing Smart Charging topology

Central Smart Charging 中央智能充电

本节内容丰富.

在中央智能充电的情况下，每个交易对充电计划的限制都是由中央系统决定的。中央系统使用这些时间表，以保持任何外部系统规定的范围内。中央系统直接控制充电点连接器的限制。

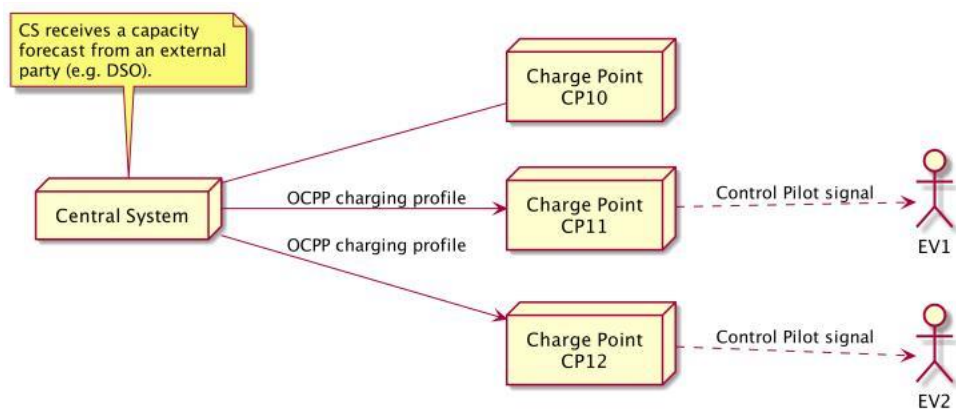


Figure 7. Central Smart Charging topology

中央智能充电假设充电限制由中央系统控制。中央系统以一种或另一种形式接收来自电网运营商(DSO)或另一来源的容量预测，并计算某些或所有充电事务的充电时间表，其详细信息超出了本规范的范围。

中央系统对连接器施加充电限制。针对Start Transaction.req PDU，中央系统可以选择使用TX配置文件为事务设置收费限制

中央智能充电可以用控制导频信号来完成，尽管有一些限制，因为EV不能通过控制导频信号充电。与本地智能充电用例类似，连接器可以通过控制引导信号执行充电计划。如下图所示：

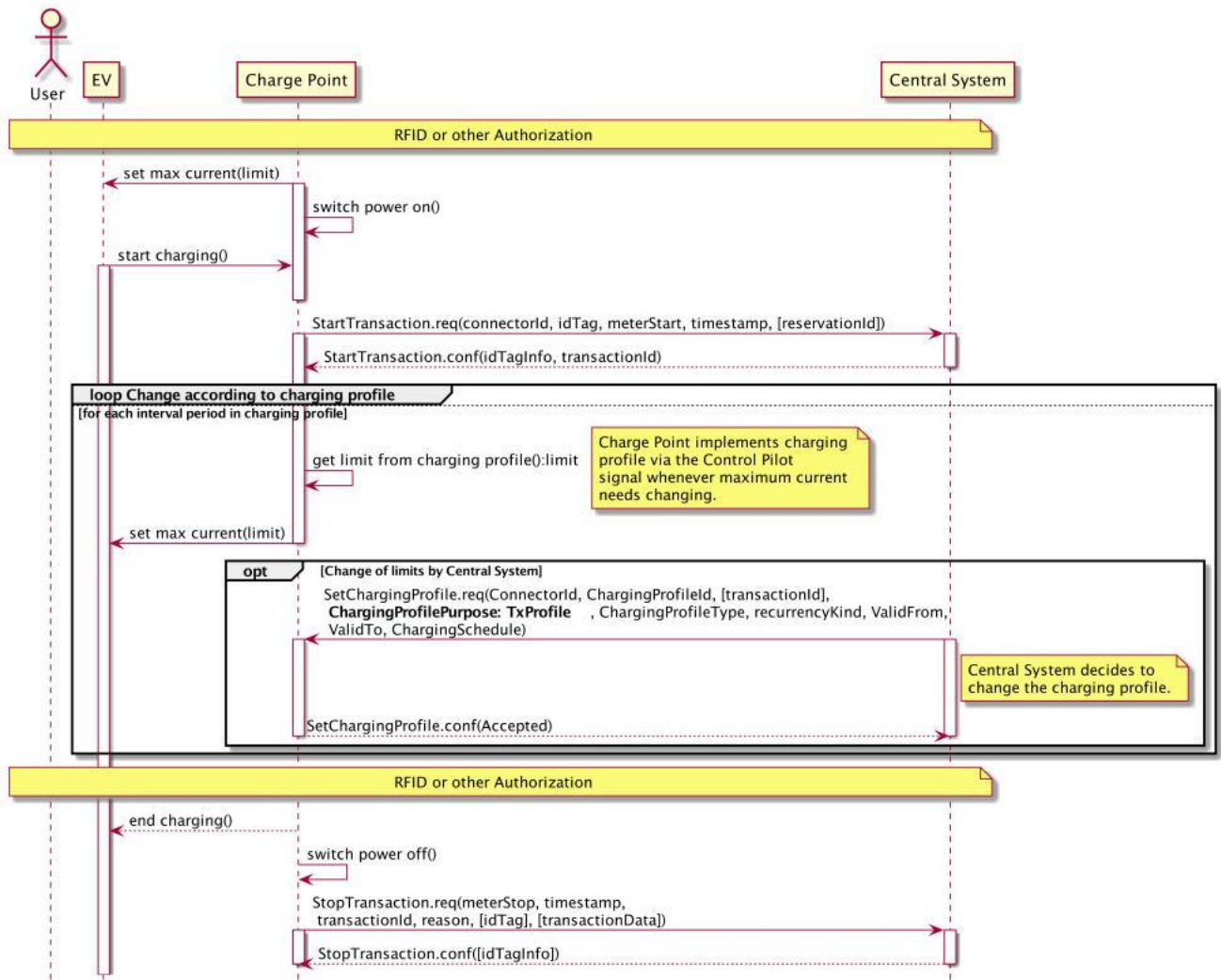


Figure 8. Sequence Diagram: Central Smart Charging

对上述图形的解释:

- 授权后，连接器将设置一个最大电流，以通过控制引导信号使用。此限制基于连接器先前从中央系统收到的（默认）充电配置文件。电动汽车开始充电，StartTransaction.req被发送到中央系统。
- 在充电过程中，连接器将根据充电配置文件不断调整最大电流或功率。可选的，在任何时间点，中央系统可以为连接器发送一个新的充电配置文件，该配置文件应用作EV的限制时间表。

Local Smart Charging 本地智能充电

本地智能充电用例描述了一个用例，其中智能充电启用的充电点具有由本地控制器而不是中央系统本地控制的充电限制。本地智能充电的用例是将一组充电点可以使用的功率限制在一定的最大值。一个典型的用途是停车库中的一些收费点，其中连接到网格的评级小于收费点的评级之和。另一个应用程序可能是本地控制器从DSO或本地智能电网节点接收有关电源可用性的信息。

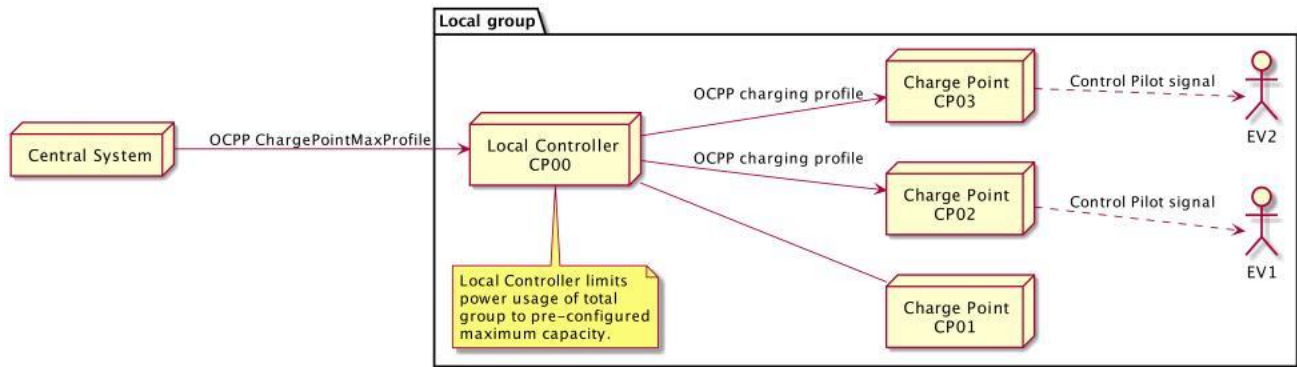


Figure 9. Local Smart Charging topology

本地智能充电假设存在一个本地控制器来控制一组电荷点。本地控制器是一个逻辑组件。它可以作为一个单独的物理组件来实现，也可以作为控制其他一些电荷点的“主”电荷点的一部分来实现。本地控件实现OCPP协议，是组成员OCPP消息的代理，并且可能有也可能没有自己的任何连接器。

在本地智能充电的情况下，本地控制器对充电点施加充电限制。这些限制可以在充电过程中动态更改，以使充电点组的功耗保持在组限制内。组限制可以在本地控制器中预先配置，也可以由中央系统配置。

可选的充电计划字段最小充电速率可由本地控制器用于优化连接器之间的功率分配。该参数通知本地控制器，低于最小充电率的充电效率低下，从而有可能选择另一种平衡策略。

下图说明了在本地智能充电组中设置充电点充电限制的消息序列。这些限制可以在本地控制器中以某种方式预先配置，也可以由中央系统设置。本地控制器包含通过根据需要调整其限制在连接的连接器的分配此容量的逻辑。

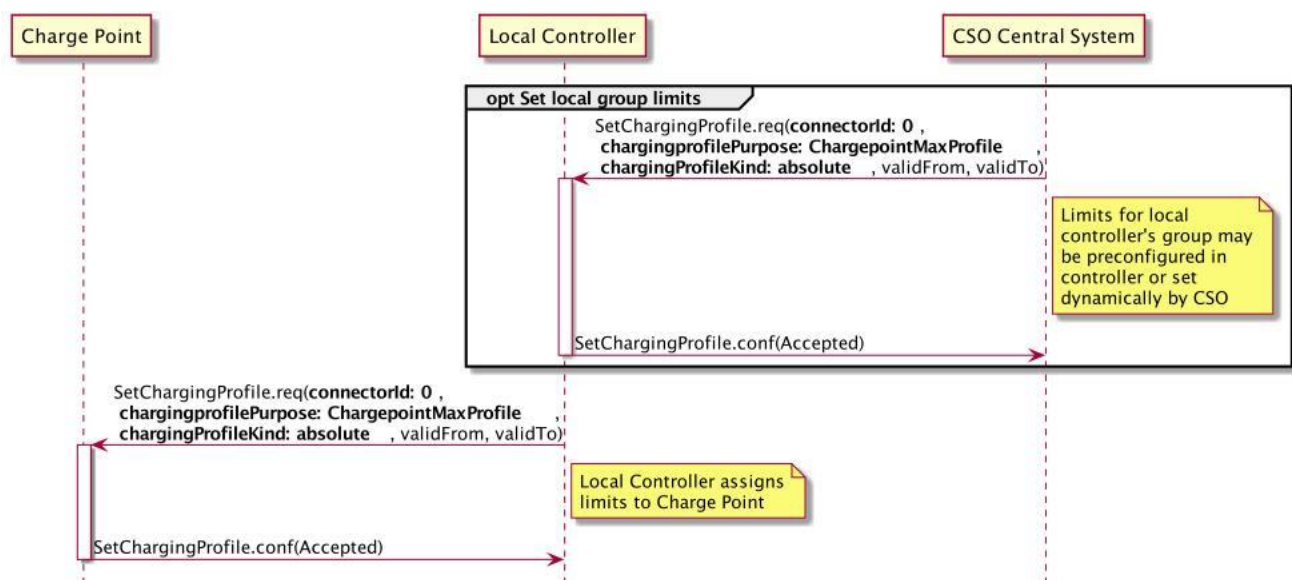


Figure 10. Presetting Local Group Limits

下一个图表描述了本地智能充电典型案例的消息序列。为了简单起见，这种情况只涉及一个连接器。

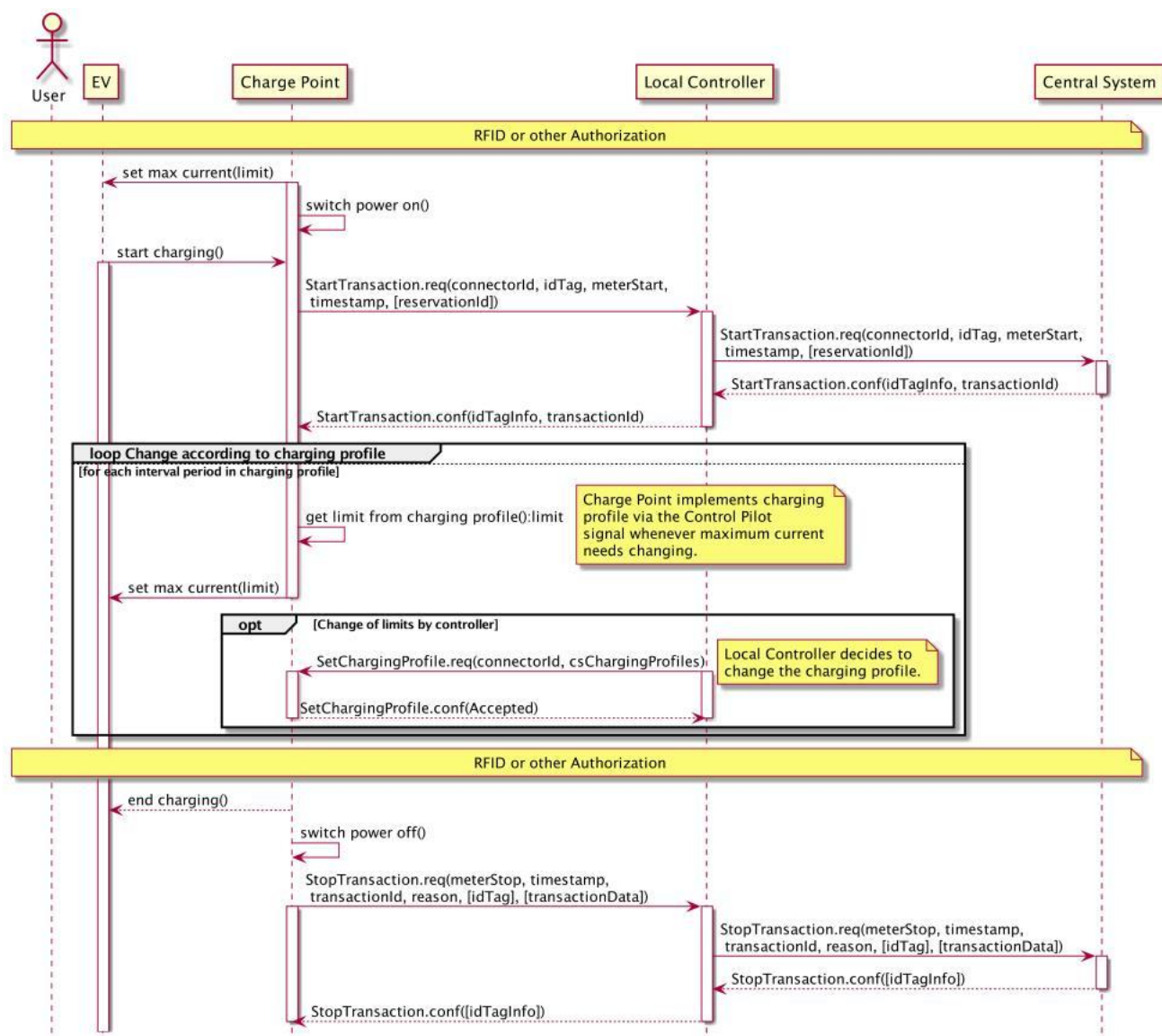


Figure 11. Sequence Diagram: Local Smart Charging

上图说明：

- 授权后，连接器将通过控制引导信号设置要使用的最大电流。此限制基于连接器先前从本地控制器收到的（默认）充电配置文件。EV启动充电并发送Start Transaction.req。
- Start Transaction.req通过本地控制器发送到中央系统，以便本地控制器也知道事务已经启动。本地控制器只传递充电点和中央系统之间的消息，以便中央系统可以单独寻址所有本地智能充电组成员
- 在充电过程中，连接器将根据充电配置文件不断调整最大电流。
可选的，在任何时间点，本地控制器可以向连接器发送新的充电配置文件，该配置文件应用作EV的限制时间表。

3.12.5. Discovery of Charge Point Capabilities 电荷点能力的发现

本节是规范性的.

定义的智能充电选项可以广泛使用。由于电荷点之间的能力可能存在限制和差异，中央系统需要能够发现电荷点的特定能力。这是由本章定义的标准化配置键来保证的。一个智能充电启用的充电点应该通过GetConfiguration.reqPDU实现和支持以下配置密钥的报告

定义的智能充电选项可以广泛使用。由于电荷点之间的能力可能存在限制和差异，中央系统需要能够发现电荷点的特定能力。这是由本章定义的标准化配置键来保证的。一个智能充电启用的充电点应该通过GetConfiguration.reqPDU实现和支持以下配置密钥的报告

智能充电配置按键
充电配置文件最大堆叠水平
充电时间表允许充电率单位
充电时间表最长周期
安装最大充电配置文件

所有标准化配置密钥的完整列表可在“标准配置密钥”一章中找到名称和价值.

3.12.6. Offline behavior of smart charging 智能充电的离线行为

本节是规范性的.

如果一个收费点在收到一个特定于交易的收费配置文件后离线TX配置文件，然后它将在事务期间继续使用此配置文件.

如果在事务启动之前或在接收到具有目的TX配置文件的特定事务收费配置文件之前，收费点离线，那么它将使用可用的收费配置文件。零或更多的以下充电配置文件的目的可能已经从中央系统收到：

* 充电点最大配置文件

* TX 默认配置文件

有关如何将充电配置文件与不同用途相结合的说明，请参阅“组合充电配置文件”一节.

如果一个充电点离线，没有任何充电配置文件，那么它应该执行一个事务，就像没有约束一样.

3.12.7. Example data structure for smart charging 智能充电的示例数据结构

本节内容丰富

下面的数据结构描述了一个每日默认配置文件，它将电源限制在08：00h到20：00h之间的6kw。

充电配置文件			
充电配置文件ID	100		
堆栈级别	0		
充电配置文件的目的	TX默认配置文件		
充电配置文件种类	反复出现		
递归类型	Daily		
收费时间表	(List of 1 充电时间表)		
	收费时间表		
	期限	86400 (= 24 hours)	
	开始时间表	2013-01-01T00:00Z	
	费率单位	W	
	收费时间表	(List of 3 收费时间表)	
		收费时间表	
		开始期	0 (=00:00)
		限制	11000
		相数	3
		起始时间	28800 (=08:00)
		limit	6000
		相数	3
		起始时间	72000 (=20:00)
		limit	11000
		相数	3

IMPORTANT

充电过程中使用的相量受到以下能力的限制：CP和EV之间的充电点、EV和电缆。如果这3个中的任何一个不能进行3相充电，则EV将只使用1相充电。

IMPORTANT

在计划或充电期间切换使用的阶段的数量应该小心进行。一些EVS可能不支持这一点，改变相量可能导致物理损伤。使用配置键：连接器开关3到1阶段支持的充电点可以判断它是否支持在事务期间切换阶段的数量。

TIP

在DST生效或失效的几天，可能需要一个特殊的配置文件(例如。用于相对剖面)。

3.13. Time zones

This section is informative.

OCPP没有规定使用特定时间值。然而，强烈建议在所有时间值中使用UTC，以提高中央系统和收费点之间的互操作性。

4. Operations Initiated by Charge Point

由收费点发起的操作

4.1. Authorize 授权

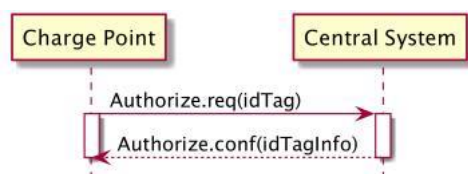


Figure 12. 顺序图：授权

在电动汽车车主开始或停止充电之前，充电点必须授权操作。收费点只应在授权后提供能源。当停止事务时，当用于停止事务的标识符与启动事务的标识符不同时，收费点将只发送一个Authorize.req。

Authorize.req应该仅用于授权用于收费的标识符。

充电点可以在本地授权标识符，而不涉及中央系统，如本地授权列表中所述。如果用户呈现的 idtag 不在本地授权列表或授权缓存中，那么充电点将发送 authorize.req pdu 到中央系统请求授权。如果 idtag 出现在本地授权列表或授权缓存中，那么充电点可以向中央系统发送 authorize.req pdu。

在收到Authorize.req PDU后，中央系统将使用Authorize.conf PDU进行响应。此响应PDU应指示idtag是否被中央系统接受。如果中央系统接受idtag，则响应PDU可能包括父idtag，并且必须包括指示接受或拒绝原因的授权状态值。

如果充电点已经实现了授权缓存，那么在收到Authorize.confPDU后，如果id标记不在本地授权列表中，则充电点将更新缓存条目，其中ID标记信息值来自授权缓存中描述的响应。

4.2. Boot Notification 启动通知

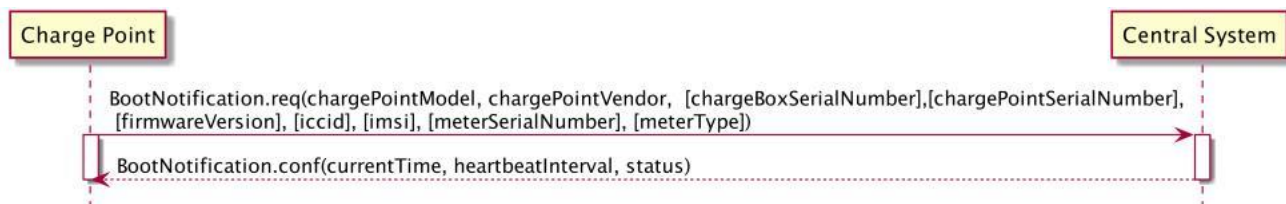


Figure 13. 序列图：启动通知

启动后，充电点应向中央系统发送请求，并提供有关其配置的信息(例如。版本、供应商等。)。中央系统应作出反应，以表明它是否接受收费点

充电点应该发送一个BootNotification.req PDU每次启动或重新启动。在物理启动/重新启动和成功完成启动通知之间，当中央系统返回接受或挂起时，充电点不应向中央系统发送任何其他请求。这包括缓存的消息，仍然存在于收费点从以前。

当中央系统响应BootNotification.conf与状态接受时，充电点将根据响应PDU的间隔调整心跳间隔，并建议将其内部时钟与提供的中央系统的当前时间同步。如果中央系统返回的东西不是接受的，则间隔字段的值指示发送下一个引导通知请求之前的最小等待时间。如果间隔值为零，充电点自行选择一个等待间隔，以避免向中央系统注入请求。充电点不应提前发送BootNotification.req，除非请求使用TriggerMessage.req这样做。

如果中央系统返回状态拒绝，充电点将不会向中央系统发送任何OCPP消息，直到上述重试间隔过期。在此期间充电点可能不再可以从中央系统到达。例如，它可能关闭其通信通道或关闭其通信硬件。中央系统也可能关闭通信通道，例如释放系统资源。当拒绝时，充电点不应响应任何中央系统启动的消息。中央系统不得启动任何。

中央系统也可能返回一个挂起的注册状态，以表明它希望在中央系统接受收费点之前检索或设置收费点的某些信息。如果中央系统返回挂起状态，则通信信道不应由充电点或中央系统关闭。中央系统可以发送请求消息，从充电点检索信息或更改其配置。充电点应该响应这些消息。充电点不应向中央系统发送请求消息，除非中央系统已指示它使用TriggerMessage.req请求这样做。

当处于挂起状态时，不允许以下中央系统启动的消息：

RemoteStartTransaction.req和RemoteStopTransaction.req

NOTE

虽然尚未被中央系统接受，但收费点可以允许本地授权的事务，如果它被配置为这样做，如本地授权和离线行为中所描述的。想要实施这种行为的各方必须认识到，这些交易是否能够交付给中央系统尚不确定

4.3. Data Transfer 数据传输



Figure 14. 序列图：数据传输

如果一个充电点需要向中央系统发送 ocpp 不支持的功能的信息，它应该使用 `datatransfer.req pdu`。

请求中的供应商ID应该为中央系统所知，并唯一地标识特定于供应商的实现。供应商ID应该是来自反向DNS名称空间的值，其中名称的顶层在反向时应该对应于供应商组织的公开注册的主DNS名称。

可选地，请求PDU中的消息ID可用于指示特定的消息或实现。

请求和响应PDU中的数据长度未定义，应由所有相关缔约方商定。

如果请求的接收方没有特定供应商ID的实现，则应返回状态“未知供应商”，数据元素不存在。如果消息ID不匹配（如果使用），收件人将返回状态“未知消息ID”。在所有其他情况下，使用状态‘接受’或‘拒绝’和数据元素是所涉各方之间特定于供应商的协议的一部分。

4.4. Diagnostics Status Notification 诊断状态通知

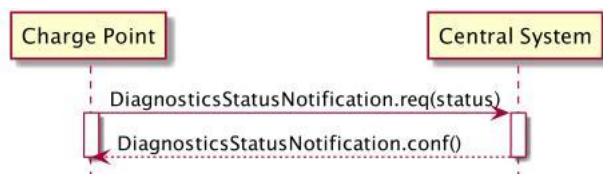


Figure 15. 序列图：诊断状态通知

充电点发送通知，通知中央系统有关诊断上传的状态。充电点应发送诊断状态通知。`reqPDU`通知中央系统诊断上传繁忙或已成功或失败。充电点只应在收到诊断状态通知的触发消息后发送状态空闲，当它不忙着上传诊断时。

在收到诊断状态通知`reqPDU`后，中央系统将使用诊断状态通知`conf`进行响应。

4.5. Firmware Status Notification 固件状态通知



Figure 16. 顺序图：固件状态通知

充电点发送通知通知中央系统固件更新的进度。充电点应发送固件状态通知。reqPDU通知中央系统下载和安装固件更新的进度。充电点只应在收到固件状态通知的触发消息后发送状态空闲，当它不忙着下载/安装固件时。

在收到固件状态通知.reqPDU后，中央系统将使用固件状态通知.conf进行响应。

4.6. Heartbeat 心跳

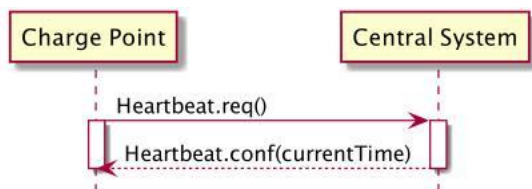


Figure 17. 顺序图：心跳

为了让中央系统知道充电点仍然连接，充电点在可配置的时间间隔后发送心跳。

充电点应该发送一个Heartbeat.req PDU，以确保中央系统知道一个充电点仍然活着。

在收到Heartbeat.req PDU后，中央系统将使用Heartbeat.conf响应.响应PDU应包含中央系统的当前时间，该时间被建议由充电点用于同步其内部时。

当另一个PDU已在配置的心跳间隔内发送到中央系统时，充电点可能会跳过发送Heartbeat.reqPDU。这意味着当一个PDU被接收时，一个中央系统应该假设一个电荷点的可用性，就像它收到一个Heartbeat.reqPDU时一样。

NOTE

通过Web Socket发送JSON，发送心跳不是强制性的。然而，对于时间同步，建议每24小时至少发送一个心跳。

4.7. Meter Values 仪表值

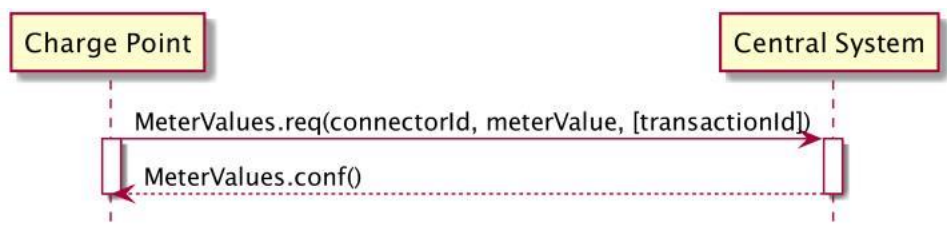


Figure 18. 顺序图：仪表值

充电点可以对能量计或其他传感器/换能器硬件进行采样，以提供关于其仪表值的额外信息。由充电点决定何时发送仪表值。可以使用 `Configuration.req` 消息更改为数据采集间隔，并指定要获取和报告的数据。

充电点应发送一个 `MeterValues.req` PDU 用于卸载仪表值。请求 PDU SHALL 包含每个示例：

- 1. 从连接器中提取样品的id。如果连接器ID为0，则与整个充电点相关联。如果连接器ID为0，并且被测量的是与能量有关的，则应从主能量计中提取样品。
- 2. 如果适用，与这些值相关的事务的事务ID。如果没有正在进行的事务，或者如果这些值是从主计数器中提取的，则可以省略事务id。
- 3. 一个或多个表值元素，类型为表值，每个元素表示在特定时间点获取的一组或多个数据值。

每个 `MeterValue` 元素包含一个时间戳和一组一个或多个单个采样值元素，这些元素都是在同一时间点捕获的。每个采样值元素包含一个值基准。每个采样值的性质由可选的度量、上下文、位置、单元、相位和格式字段决定。

可选的度量字段指定被测量/报告的值的类型。

可选上下文字段指定触发读取的原因/事件。

可选位置字段指定测量的位置(例如。入口，出口)。

可选相位字段指定该值应用于电气安装的哪个阶段或阶段。充电点应从功率计（或没有电网连接时）的角度报告所有相号相关值。

NOTE	相场不适用于所有测量。
NOTE	两个度量(当前。提供和权力。提供)是可用的，严格地说，没有测量值。它们表示电流/功率的最大数量这是提供给电动汽车，并打算用于智能充电应用。

对于单个连接器相位旋转信息，中央系统可以通过GET配置查询充电点上的连接器相位旋转配置键。充电点应报告相对于网格连接的相位旋转。每个连接器的可能值是：不适用、未知、RST、RTS、SRT、STR、TRS和TSR。有关更多信息，请参见标准配置键名和值部分。

专家可选格式字段指定数据是以普通（默认）形式表示为简单数值("Raw")，还是以“签名数据”表示，这是一个不透明的数字签名二进制数据块，表示为十六进制数据。当提供更成熟的解决方案替代方案时，这个实验领域可能会被废弃，随后在以后的版本中被删除。

为了保持向后兼容性，采样值元素上所有可选字段的默认值都是这样的，没有任何附加字段的值将被解释为以Wh（瓦特小时）单位为单位的活动导入能量的寄存器读取。

在收到Meter values.req PDU后，中央系统将使用Meter values.conf响应。

中央系统很可能会对它接收到的Meter Values.req中包含的数据进行卫生检查。这种理智检查的结果绝不会导致中央系统不能用Meter Values.conf响应。如果无法用Meter Values.conf响应，则只会导致收费点再次尝试与事务相关消息的错误响应中指定的相同消息。

4.8. Start Transaction 开始交易

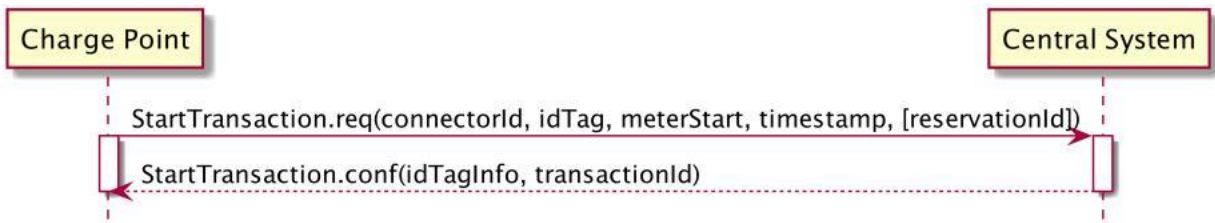


Figure 19. 序列图：开始交易

充电点将发送一个StartTransaction.req PDU到中央系统，以通知已启动的事务。如果此事务结束预订（请参阅“立即预订”操作），则Start Transaction.req必须包含预订ID。

在收到Start Transaction.req PDU后，中央系统应该用Start Transaction.conf PDU响应。此响应PDU必须包括事务id和授权状态值。

中央系统必须验证Start Transaction.req PDU中标识符的有效性，因为标识符可能是由收费点使用过时的信息在本地授权的。例如，标识符可能已经被阻塞，因为它被添加到收费点的授权缓存中。

如果充电点已经实现了授权缓存，那么在收到StartTransaction.confPDU后，如果id标签不在本地授权列表中，则充电点将更新缓存条目，其中ID标签信息值来自授权缓存中描述的响应。

中央系统很可能会对接收到的Start Transaction.req中包含的数据进行健全检查。这种理智检查的结果绝不会导致中央系统不能用Start Transaction.conf响应。如果没有使用Start Transaction.conf响应，则只会导致收费点再次尝试与事务相关消息的错误响应中指定的相同消息。

4.9. Status Notification 状态通知

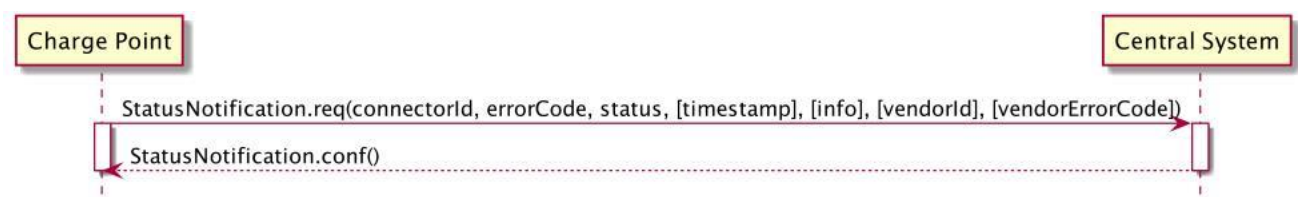


Figure 20. 序列图：状态通知

收费点向中央系统发出通知，通知中央系统状态更改或充电点内的错误。下表描述了从以前的状态（左列）到新状态(上行) 的更改，充电点可能会向中央系统发送Status Notification.req PDU。

IMPORTANT	以前的OCPP版本中定义的被占领状态不再相关。被占领国家分为五个新的地位：准备充电，暂停电动汽车，暂停电动汽车和整理。
NOTE	EVSE用于Status 通知代替套接字或 未来兼容性的充电点。

下表描述了哪些状态转换是可能的:

		1 Avail able	2 Prepa ring	3 Charg ing	4 Suspe nded EV	5 Suspe nded EVSE	6 Finish ing		7 Reser ved	8 Unav ailabl e	9 Fault ed
A	Available 可用		A2	A3	A4	A5			A7	A8	A9
B	Preparing准备工作	B1		B3	B4	B5					B9
C	Charging 充电	C1			C4	C5	C6			C8	C9
D	SuspendedEV	D1		D3		D5	D6			D8	D9
E	暂停电动汽车 SuspendedEV SE	E1		E3	E4		E6			E8	E9
F	Finishing 完成	F1	F2							F8	F9
G	Reserved 预留	G1	G2							G8	G9
H	Unavailable不可用	H1	H2	H3	H4	H5					H9
I	Faulted 有错	I1	I2	I3	I4	I5	I6		I7	I8	

NOTE

上表仅适用于连接器ID>0。对于连接器ID0，只适用有限的集合，即：可用、不可用和故障。

下一个表描述可能导致状态更改的事件:

	Not possible
A2	开始使用(例如。插入插头，海湾占用检测，呈现idTag，按下启动按钮，接收RemoteStartTransaction.req)
A3	可以在充电点没有授权的手段
A4	类似于A3，但EV不开始充电
A5	类似于A3，但EVSE不允许充电
A7	接收到保留连接器的“立即准备”消息
A8	接收到更改可用性消息，该消息将连接器设置为不可用
A9	检测到一个故障，以防止进一步的充电操作

B1	意图使用结束(例如。 插头移除, 海湾不再占用, 第二次显示idTag, 超时预期用户操作)
B3	所有充电的先决条件都得到满足, 充电过程开始
B4	充电的所有先决条件都得到满足, 但EV不开始充电
B5	所有充电的先决条件都得到满足, 但EVSE不允许充电
B9	检测到一个故障, 以防止进一步的充电操作
C1	充电会话结束时不需要用户操作(例如。 固定电缆在EV侧被移除)
C4	充电停止在EV请求(例如。 打开S2)
C5	充电停止在EVSE请求(例如。 智能充电限制, 事务被启动事务.conf中的授权状态失效)
C6	充电会话由用户或远程停止事务消息停止, 并需要进一步的用户操作(例如。 拆下电缆, 离开停车舱)
C8	充电会话结束, 不需要用户操作, 并且连接器计划成为不可用
C9	检测到一个故障, 以防止进一步的充电操作
D1	充电会话结束时, 不需要用户操作
D3	充电恢复应电动汽车的要求(例如。 S2关闭)
D5	充电由EVSE暂停(例如。 由于智能充电限制)
D6	充电会话停止, 需要进一步的用户操作
D8	充电会话结束, 不需要用户操作, 并且连接器计划成为不可用

D9	检测到一个故障，以防止进一步的充电操作
E1	充电会话结束时，不需要用户操作
E3	充电恢复，因为EVSE限制被解除
E4	取消EVSE限制，但EV不开始充电
E6	充电会话停止，需要进一步的用户操作
E8	充电会话结束，不需要用户操作，并且连接器计划成为不可用
E9	检测到一个故障，以防止进一步的充电操作
F1	所有用户操作完成
F2	用户重新启动充电会话(例如。重新连接，再次呈现id标记)
F8	所有用户操作完成，连接器计划成为不可用
F9	检测到一个故障，以防止进一步的充电操作
G1	预订到期或收到取消预订消息
G2	保留身份显示
G8	预订到期或收到取消预订消息，连接器将成为不可用
G9	检测到一个故障，以防止进一步的充电操作
H1	连接器由更改可用性消息设置为可用
H2	在用户与充电点交互后，可设置连接器

H3	连接器被设置为可用的，并且不需要用户操作来开始充电
H4	类似于H3，但EV不启动字符
H5	类似于H3，但EVSE不允许充电
H9	检测到一个故障，以防止进一步的充电操作
I1-I8	故障解决，状态恢复到故障前状态

IMPORTANT	充电点连接器可能有9种状态中的任何一种，如上表所示。对于连接器ID0，只适用有限的集合，即：可用、不可用和故障。连接器ID0的状态与个别连接器的状态没有直接连接(>0)。
IMPORTANT	如果充电被EV和EVSE暂停，状态挂起的EVSE将优先于状态挂起的EVSE。
IMPORTANT	当一个充电点或连接器被设置为“更改可用性”命令无法使用的状态时，“不可用”状态 必须在重新启动时保持持久性。充电点可能会在内部将不可用状态用于其他目的(例如。同时更新固件或等待初始接受注册状态)。

由于被占领的状态已经被划分为五个新的状态(准备，充电，暂停EV，暂停EVSE和整理)，更多的状态通知。例如当一个事务启动时，连接器状态将从准备短暂的暂停/开发和/或暂停/开发中间连接，可能在几秒钟之内连接器状态发生变化。

为了限制转换的数量，充电点可能会忽略发送StatusNotification.req，如果它的活动时间小于可选配置键最小状态持续时间中定义的时间。这样，收费点可能选择不发送某些StatusNotification.reqPDU。

NOTE	充电点制造商可能已经实现了一个最低状态持续时间的某些状态转换，单独的最低状态持续时间设置最小状态持续时间中设置的时间将添加到此默认延迟中。将最小状态持续时间设置为零，不应覆盖默认制造商的最小状态持续时间
IMPORTANT	设置较高的最低状态持续时间可能会导致延迟发送所有状态通知，因为充电点只会在通过最低状态持续时间后发送状态通知

充电点可能会发送StatusNotification.req PDU通知中央系统故障条件。当“状态”字段没有故障时，该条件应被视为警告，因为充电操作仍然是可能的。

IMPORTANT

充电点错误代码EV通信错误仅用于状态准备、挂起EV、挂起EVSE和整理，并被视为警告。

当充电点配置为EV侧断开设置为false的停止事务时，事务正在运行，EV在EV侧断开连接，然后将StatusNotification.req发送到中央系统，将“错误代码”字段设置为：“无错误”。充电点应在“信息”字段中添加其他信息，以暂停的原因通知中央系统：“EV侧断开”。当前事务没有停止。

当充电点配置为EV侧断开设置为true的停止事务时，事务正在运行，EV在EV侧断开连接，然后将状态通知.req发送到中央系统，将“错误代码”字段设置为：“无错误”。充电点应该在“信息”字段中添加其他信息，以停止的原因通知中央系统：“EV侧断开”。当前事务停止。

当充电点在离线后连接到中央系统时，它根据以下规则更新中央系统的状态：

1. 充电点应该发送一个StatusNotification.req PDU与其当前状态，如果状态改变，而充电点是离线的。
2. 充电点可能会发送StatusNotification.req PDU报告在充电点离线时发生的错误。
3. 充电点不应发送StatusNotification.req PDU用于在充电点离线时发生的历史状态更改事件，并且不通知中心系统的充电点错误或充电点的当前状态。
4. StatusNotification.req 消息必须按照它们所描述的事件发生的顺序发送。

在收到StatusNotification.req PDU后，中央系统将使用StatusNotification.conf PDU进行响应。

4.10. Stop Transaction 停止交易

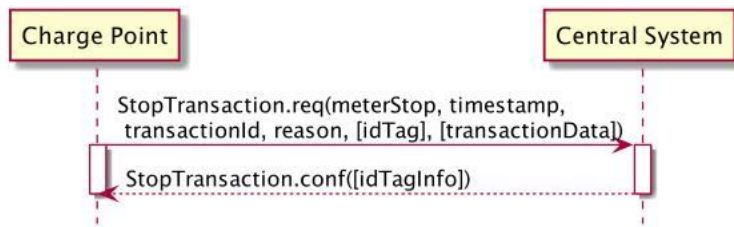


Figure 21. 序列图：停止事务

当事务停止时，充电点将发送StopTransaction.reqPDU，通知中央系统事务已停止。

StopTransaction.req PDU可能包含一个可选的事务数据元素，以提供有关事务使用的更多细节。可选事务数据元素是任意数量的电表值的容器，使用与MeterValues.req PDU的表值元素相同的数据结构（参见部分表值）

在收到StopTransaction.req PDU后，中央系统将使用StopTransaction.confPDU响应。

NOTE

中央系统不能阻止事务停止。它可能只通知充电点，它已经收到StopTransaction.req和可能发送有关用于停止事务的id标记的信息。如果已实现，则应使用此信息更新授权缓存。

请求PDU中的idTag在 充电点本身需要停止交易。例如，当要求充电点重置时。

如果事务以正常方式结束(例如。EV-driver提出了他的身份来阻止交易)，原因元素可能被省略，原因应该被假定为“本地”。如果事务没有正常结束，则应将原因设置为正确的值。作为正常交易终止的一部分，充电点将解锁电缆（如果不是永久附加的）。

充电点可以解锁电缆（如果不是永久连接）时，电缆断开在 EV。如果支持，此功能将由EVSide断开上的配置键解锁连接器报告和控制。

当电缆在EV上断开时，充电点可能会解锁电缆(如果不是永久连接的话。如果支持，此功能将由EV侧断开上的配置键解锁连接器报告和控制。

如果EVSideDisconnect上的停止事务设置为false，则当电缆与EV断开时，事务不应停止。如果EV重新连接，则允许再次进行能量转移。在这种情况下，没有机制防止其他EVS在同一正在进行的事务中充电和断开连接。在EVSide断开连接设置为false时，连接器将保持锁定在充电点，直到用户显示标识符。

通过将EVSide断开连接上的停止事务设置为true，当电缆与EV断开时，事务将被停止。如果 电动汽车被重新连接，能量转移是不允许的，直到交易停止和一个新的交易开始。如果EVSide断开上的解锁连接器设置为true，则充电点上的连接器也将被解锁。

NOTE 如果EVSide断开连接上的停止事务设置为false，则为 在EVSide断开时，应优先于解锁连接器。换句话说：当电缆在EV侧断开时，电缆总是保持锁定，当EVSide断开上的停止事务为false时。

NOTE 设置停止交易在EV侧断开为真将防止破坏行为顶部停止能量流，通过拔出未锁定的电缆在EV侧。

中央系统很可能会对它接收到的Stop Transaction.req中包含的数据进行理智检查。这种理智检查的结果绝不会导致中央系统不能用StopTransaction.conf响应。如果无法使用StopTransaction.conf响应，则只会导致收费点再次尝试与事务相关消息的错误响应中指定的相同消息。

如果充电点已经实现了授权缓存，那么在收到StopTransaction.conf PDU后，如果idTag不在本地授权列表中，则充电点将更新缓存条目，其中IDTag信息值来自授权缓存中描述的响应。

5. Operations Initiated by Central System

由中央系统发起的操作

5.1. Cancel Reservation 取消预订



Figure 22. 顺序图：取消预订

要取消预订，中央系统将发送一个CancelReservation.req PDU到收费点。

如果收费点有一个与请求PDU中的预订ID相匹配的预订，它将返回状态‘接受’。否则，它将返回“拒绝”。

5.2. Change Availability 更改可用性



Figure 23. 序列图: 更改可用性

中央系统可以请求一个充电点来改变它的可用性。充电点在充电或准备充电时被认为是可用的(“操作”。当充电点不允许任何充电时，它被认为是不可用的。中央系统应该发送一个ChangeAvailability.req PDU请求一个充电点来改变它的可用性。中央系统可以将可用性更改为可用或不可用。

在收到ChangeAvailability.req PDU后，充电点将使用ChangeAvailability.conf PDU响应。响应PDU应指示充电点是否能够更改到请求的可用性。当事务正在进行中时，充电点应以“预定”的可用性状态响应，以指示它计划在事务完成后发生。

如果中央系统请求充电点更改为它已经处于的状态，充电点将以可用性状态“接受”响应’。

当使用ChangeAvailability.req PDU请求的可用性更改发生时，充电点应将其新的可用性状态通知中央系统，并提供状态通知。

NOTE 在ChangeAvailability.req包含连接器ID=0的情况下，状态更改适用于充电点和所有连接器。

NOTE 持久状态：例如：设置为不可用的连接器将持续重新启动。

5.3. Change Configuration 更改配置



Figure 24. 顺序图: 更改配置

中央系统可以请求一个充电点来更改配置参数。为了实现这一点，中央系统应该发送一个ChangeConfiguration.req。此请求包含一个键值对，其中“键”是要更改的配置设置的名称，“值”包含配置设置的新设置。

在收到ChangeConfiguration.req充电点后，应使用ChangeConfiguration.conf回复，指示它是否能够执行更改没有规定“键”和“价值”的内容。如果“键”不对应于充电点支持的配置设置，则应以未支持的状态回复。如果更改成功执行，充电点将以接受的状态响应。如果更改已成功执行，但需要重新启动才能应用它，则充电点将响应状态重新启动。如果未能设置配置，充电点将响应状态拒绝。

如果一个键值被定义为CSL，它可能伴随着一个[键名称]最大长度键，指示项目中CSL的最大长度。如果未设置此键则应假定1（1）项的安全值。

5.4. Clear Cache 清除缓存

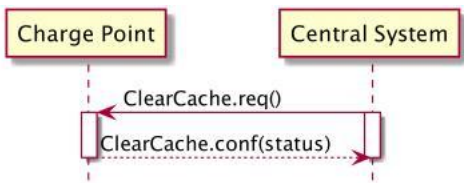


Figure 25. 序列图：清除缓存

中央系统可以请求一个收费点来清除其授权缓存。中央系统应发送一个清晰的Cache.reqPDU，以清除充电点的授权缓存。

在收到ClearCache.reqPDU后，充电点将用ClearCache.confPDU响应。响应PDU应指示充电点是否能够清除其授权缓存。

5.5. Clear Charging Profile 清除的充电配置文件

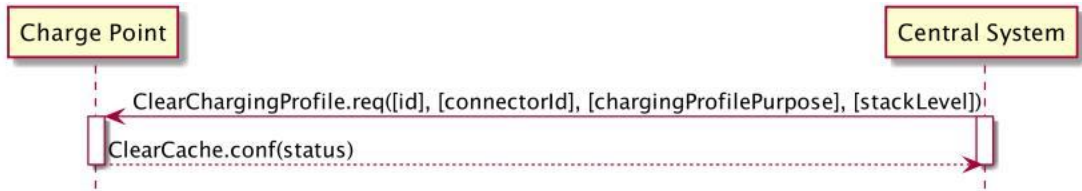


Figure 26. 顺序图：清除充电配置文件

如果中央系统希望清除先前发送的充电点的部分或全部充电配置文件，则应使用ClearChargingProfile.req PDU。

充电点应该用一个清晰的ClearChargingProfile.conf PDU指定它是否能够处理请求。

5.6. Data Transfer 数据传输

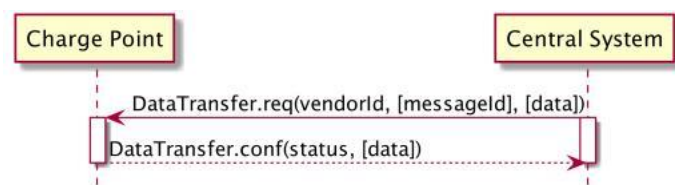


Figure 27. 序列图：数据传输

如果中央系统需要为OCPP不支持的函数向充电点发送信息，它将使用DataTransfer.req PDU.

此操作的行为与电荷点发起的数据传输操作相同。有关详细信息，请参阅数据传输.

5.7. Get Composite Schedule 得到综合时间表

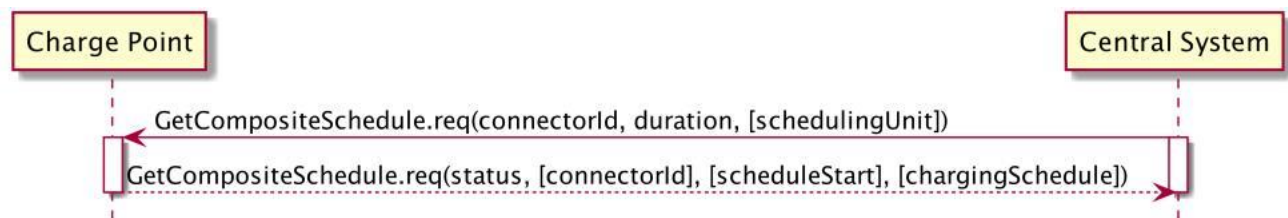


Figure 28. 顺序图：获得复合时间表

中央系统可能会请求收费点通过发送GetCompositeSchedule.req PDU时间表。所报告的时间表 GetCompositeSchedule.conf PDU，是计算所有活动计划和充电点中可能存在的局部限制的结果。此外，IEC15118的限制也可以考虑在内.

在收到GetCompositeSchedule.req后，充电点应计算到持续时间的预定时间间隔，并将它们发送到中央系统.

如果请求中的连接器ID设置为‘0’，则充电点应报告所要求的时间段内充电点的总预期能量流.

NOTE

请注意，充电点发送的充电时间表只是对该时间点的指示。由于外部原因，此时间表可能会随着时间的推移而改变（例如，基于网格连接容量的本地平衡是活动的，并且一个连接器可用）。

如果充电点无法报告所请求的时间表，例如，如果连接器ID未知，它将以拒绝的状态响应.

5.8. Get Configuration 获取配置

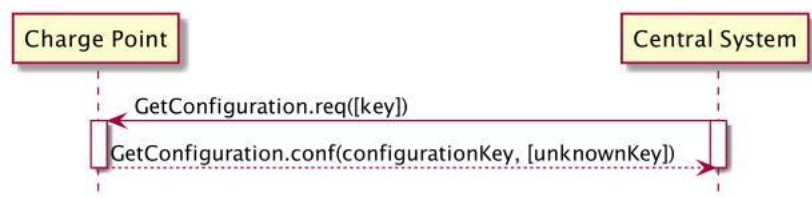


Figure 29. 序列图：获取配置

要检索配置设置的值，中央系统将向充电点发送GetConfiguration.reqPDU。

如果请求PDU中的键列表为空或丢失（它是可选的），则充电点将返回GetConfiguration.conf中所有配置设置的列表。否则，充电点将返回一个已识别的键列表及其相应的值和只读状态。未识别的密钥应作为GetConfiguration.conf的可选未知密钥列表元素的一部分放置在响应PDU中。

单个PDU中请求的配置键数可能受到充电点的限制。这个最大值可以通过读取配置键获取配置Max键来检索。

5.9. Get Diagnostics 获取诊断

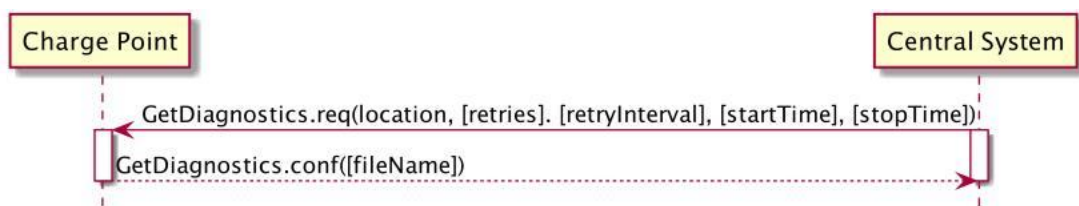


Figure 30. 序列图：获取诊断

中央系统可以请求充电点提供诊断信息。中央系统应该发送一个GetDiagnostics.reqPDU，以获得充电点的诊断信息其中充电点必须将其诊断数据上传到请求的诊断信息的开始和结束时间。

收到GetDiagnostics.req PDU后，如果诊断信息可用，则充电点将用GetDiagnostics.conf PDU响应，说明包含将上载的诊断信息的文件的名称。充电点应上传一个文件。诊断文件的格式没有规定。如果没有诊断文件可用，那么GetDiagnostics.conf将不包含文件名。

5.10. Get Local List Version 获取本地列表版本



Figure 31. 序列图: 获取本地列表版本

为了支持本地授权列表的同步，中央系统可以请求一个本地授权列表版本号的收费点。中央系统应该发送一个GetLocalListVersion.reqPDU来请求这个值。

收到GetDiagnostics.reqPDU后，如果诊断信息可用，则充电点将用GetDiagnostics.conf PDU响应，说明包含将上载的诊断信息的文件的名称。充电点应上传一个文件。诊断文件的格式没有规定。如果没有诊断文件可用，那么GetDiagnostics.conf将不包含文件名。

5.11. Remote Start Transaction 远程启动事务



Figure 32. 序列图: 远程启动事务

中央系统可以通过发送RemoteStartTransaction.req请求收费点启动事务。收到后，充电点将使用RemoteStartTransaction.conf和指示它是否能够启动事务的状态进行回复。

RemoteStartTransaction.req消息的效果取决于在收费点中授权远程TX请求配置密钥的值52

- 如果授权远程TX请求的值为真，则充电点将表现为响应在 收费 指向以RemoteStartTransaction.req消息中给出的idTag启动事务。这意味着收费点将首先尝试使用本地授权列表、授权缓存和/或Authorize.req请求来授权idTag。只有在获得授权后才会启动事务。
- 如果授权远程TX请求的值为false，则充电点将立即尝试为RemoteStartTransaction.req消息中给出的idTag启动事务。注意，事务启动后，收费点将向中央系统发送启动事务请求，中央系统在处理此启动事务请求时将检查idTag的授权状态。

以下典型用例是远程启动事务的原因：

- 启用CPO操作符帮助启动事务有问题的EV驱动程序。
- 使移动应用程序能够通过中央系统控制收费交易。
- 启用SMS以通过中央系统控制收费事务。

RemoteStartTransaction.req应该包含一个标识符(idTag)，如果它能够启动事务，则充电点将使用该标识符(idTag)将StartTransaction.req发送到中央系统。事务的启动方式与Start Transaction中描述的相同。 如果要在特定连接器上启动事务，则RemoteStartTransaction.req 可能包含连接器id。当没有提供连接器id时，充电点控制连接器选择。充电点可能会拒绝没有连接器id的RemoteStartTransaction.req。

中央系统可能在远程启动事务请求中包含一个充电配置文件。此充电配置文件的目的应设置为TX配置文件。如果接受，收费点将使用此收费配置文件进行交易。

NOTE

如果没有支持智能充电的充电点接收到带有充电配置文件的RemoteStartTransaction.req，则该参数将被忽略。

5.12. Remote Stop Transaction 远程停止事务



Figure 33.序列图: 远程停止事务

中央系统可以通过发送具有事务标识符的RemoteStopTransaction.req到收费点来请求收费点停止事务。充电点应回复RemoteStopTransaction.conf，以指示它是否确实能够停止事务。

停止事务的远程请求等于停止事务的本地操作。因此，事务应该停止，充电点应该发送一个StopTransaction.req，如果适用的话，解锁连接器。

以下两个主要用例是远程停止事务的原因：

- 启用CPO操作符，以帮助存在问题的EV驱动程序停止事务。
- 使移动应用程序能够通过中央系统控制收费交易。.

5.13. Reserve Now 预约

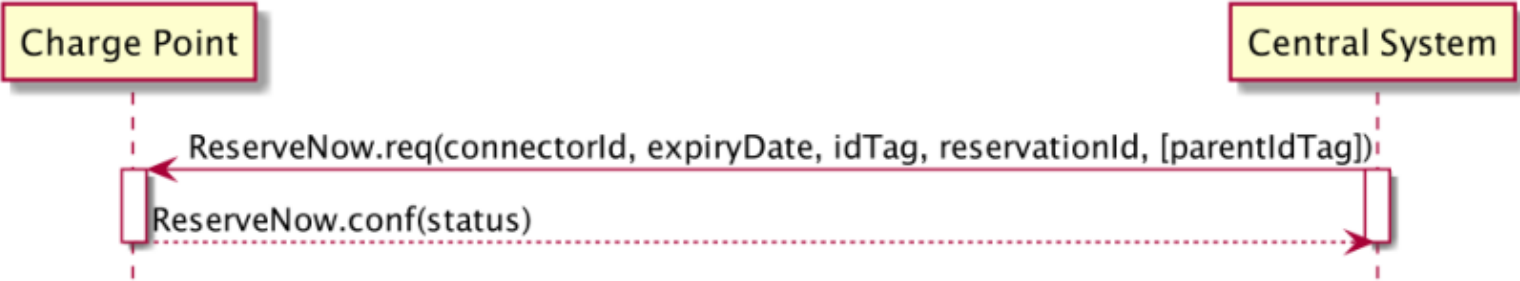


Figure 34. 顺序图：预约

中央系统可以向充电点发出ReserveNow.req，以预留连接器供特定idTag使用。

若要请求预订，中央系统将立即发送备用电源到充电点。中央系统可以指定要保留的连接器。当收到一个ReserveNow.req PDU，充电点将响应一个ReserveNow.conf PDU。

如果请求中的预订ID与收费点中的预订相匹配，则收费点将用请求中的新预订替换该预订。

如果预订ID与充电点中的任何预订不匹配，那么充电点将返回状态值‘接受’，如果它成功地保留了连接器。如果充电点或指定的连接器被占用，则充电点将返回“已占用”。当充电点或连接器已为相同或另一个id标签保留时，充电点也应返回“已占用”。如果充电点或连接器处于故障状态，则充电点将返回“故障”。如果充电点或连接器处于不可用状态，则充电点将返回“不可用”。充电点将返回“拒绝”，如果它被配置为不接受预订。

如果充电点接受预订请求，则将拒绝对保留连接器上的所有传入idTag进行充电，除非传入idTag或父idTag与预订的idTag或父idTag匹配。

当配置键：ReserveConnectorZeroSupport设置为true时，充电点支持对连接器0的保留。如果预订请求中的连接器ID为0，则充电点不应保留特定的连接器，但应确保在预订有效期内的任何时候，保留一个连接器可用于保留IDTag。如果配置键：保留连接器零支持没有设置或设置为false，充电点将返回“拒绝”。

如果保留中的父id标记有一个值（它是可选的），那么为了确定与传入idTag相关联的父idTag，收费点可能会在其本地授权列表或授权缓存中查找它。如果在本地授权列表或授权缓存中找不到它，则充电点将向中央系统发送传入idTag的Authorize.req。Authorize.conf响应包含父标识。

当（1）为保留idTag或父idTag启动事务时，在充电点上，以及当保留连接器ID为0时，在预留连接器或任何连接器上，或（2）当达到过期日期指定的时间时，或（3）当充电点或连接器设置为故障或不可用时，预订将终止。

如果已启动保留idTag的事务，则充电点将在StartTransaction.reqPDU(参见StartTransaction)中发送预订ID，以通知中央系统预订已终止。

当预订期满时 充电点应终止预订，并使连接器可用。充电点应发送状态通知，通知中央系统保留连接器现在可用。

如果充电点已经实现了授权缓存，那么在收到ReserveNow.conf PDU后，如果idTag不在本地授权列表中，则充电点将更新缓存条目，并使用来自“授权缓存”下描述的响应的IDTag信息值。

NOTE

建议在接收Reserve Now.req和事务开始之前使用authorize.req验证标识符。

5.14. Reset 重置

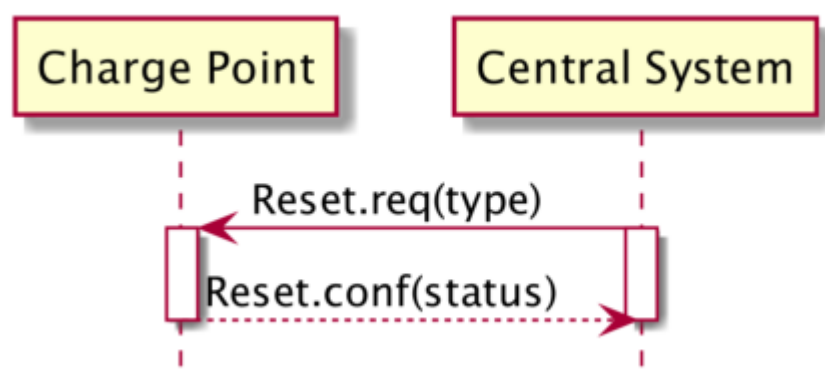


Figure 35. 顺序图: 重置

中央系统应该发送一个Reset.reqPDU，请求一个充电点来重置自己。中央系统可以请求硬复位或软复位。在收到Reset.reqPDU后，充电点将用Reset.conf PDU响应。响应PDU应包括充电点是否试图重置自己

在收到软复位，充电点将返回到一个状态，表现为刚刚启动。如果任何事务正在进行中，则应在重置之前正常终止如在停止事务中。

在收到硬重置时，充电点将尝试终止任何正在进行的事务，通常如在停止事务中，然后执行重新启动。

NOTE | 持久状态：例如：设置为不可用的连接器应持续。

5.15. Send Local List 发送本地列表

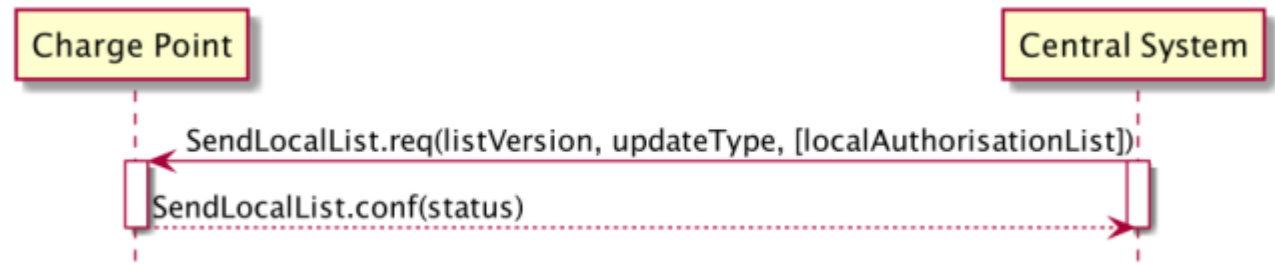


Figure 36. 序列图: 发送本地列表

中央系统可以发送一个本地授权列表，收费点可以用来授权id标签。该列表可能是一个完整的列表，以取代充电点中的当前列表，也可能是一个差异列表，其更新将应用于充电点中的当前列表。

中央系统应该发送一个SendLocalList.reqPDU将列表发送到充电点。SendLocalList.req PDU应包含更新类型（完整或差异）和版本号，收费点在更新后必须与本地授权列表关联。

在收到SendLocalList.req PDU后，收费点应以 SendLocalList.confPDU。响应PDU应指示充电点是否已接受本地授权列表的更新。如果状态失败或版本匹配错误，并且更新类型是差异的，那么中央系统应该重试发送带有更新类型Full的完整本地授权列表。

5.16. Set Charging Profile 设置充电配置文件

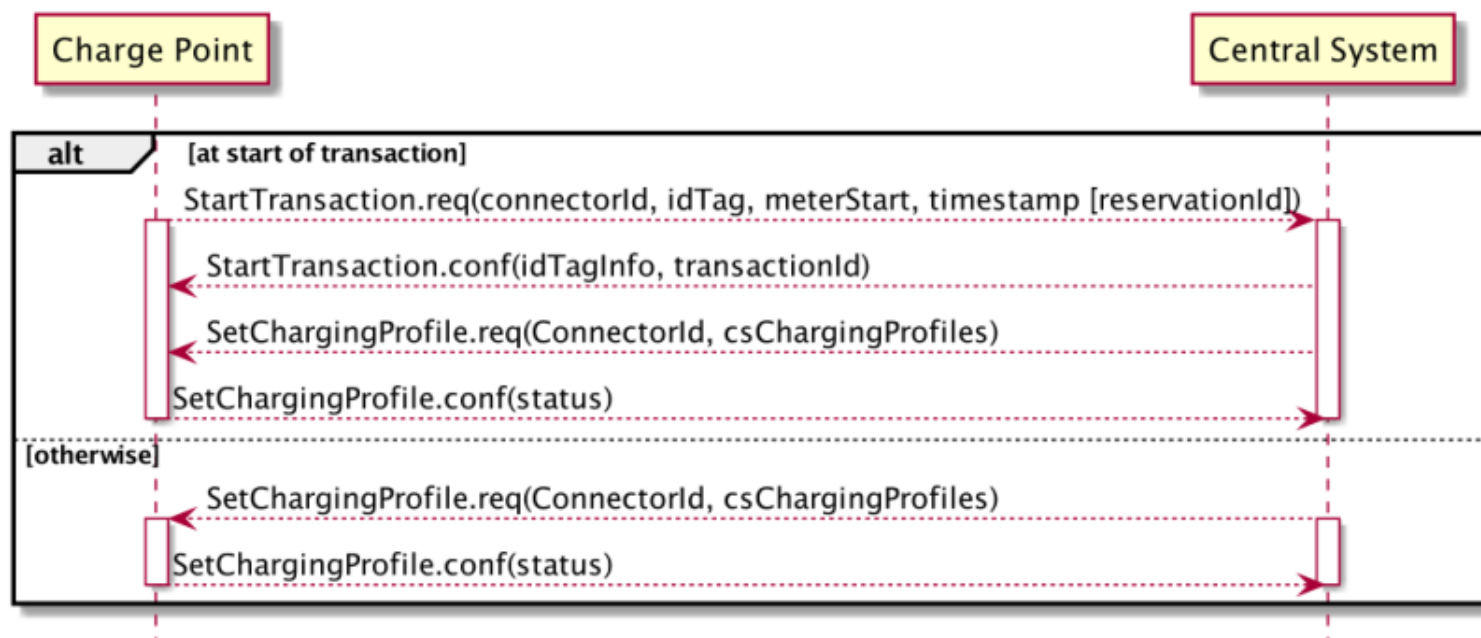


Figure 37. 顺序图：设置充电配置文件

在以下情况下，中央系统可以将SetChargingProfile.req发送到充电点，以设置充电配置文件：

- 在事务开始时设置事务的收费配置文件；
- 在发送到收费点的远程启动事务请求中
- 在事务期间更改事务的活动配置文件；
- 在事务的上下文之外，作为一个单独的消息，将充电配置文件设置为本地控制器、充电点或默认充电配置文件设置为连接器。

IMPORTANT

为了防止事务和TX配置文件之间的不匹配，如果配置文件适用于特定事务，中心系统将在SetChargingProfile.req中包含事务ID。

这些情况说明如下。

5.16.1. Setting a charging profile at start of transaction

在交易开始时设置收费配置文件

如果中央系统接收到启动事务 系统应使用Start Transaction.conf响应。 如果需要一个充电配置文件，中央系统可能会选择发送一个SetChargingProfile.req到充电点。

建议在发送收费配置文件之前检查Start Transaction.req PDU中的时间戳，以检查事务是否可能仍在进行中。 Start Transaction.req可能是在脱机期间缓存的。

5.16.2. Setting a charge profile in a RemoteStartTransaction request

在远程启动事务请求中设置收费配置文件

中央系统可能在远程启动事务请求中包含一个收费配置文件。

如果中央系统包括一个充电配置文件，充电配置文件的目的必须设置为TX配置文件。

NOTE

一旦事务被报告给中央系统，收费点应该将事务ID添加到接收到的配置文件中。

5.16.3. Setting a charging profile during a transaction. 在事务期间设置收费配置文件

中央系统可能会向充电点发送一个充电配置文件，以更新该事务的充电配置文件。中央系统应为此使用SET充电配置文件.reqPDU。如果在充电点上存在具有相同充电配置文件ID的充电配置文件，或相同的堆栈级别/充电配置文件用途组合，则新的充电配置文件将替换现有的充电配置文件，否则将添加它。然后，充电点应重新评估其收集的充电配置文件，以确定哪些充电配置文件将成为活动的。为了确保更新的充电配置文件只适用于当前事务，充电配置文件的充电配置文件目的必须设置为TX配置文件。（见章节：充电配置文件用途）

5.16.4. Setting a charging profile outside of a transaction
在事务之外设置收费配置文件

中央系统可以发送充电配置文件到一个充电点，将用作默认充电配置文件。中央系统应为此使用SET充电配置文件.reqPDU。这种充电配置文件可以随时发送。如果在充电点上存在具有相同充电配置文件ID的充电配置文件或相同的堆栈级别/充电配置文件用途组合，则新的充电配置文件将替换现有的充电配置文件，否则将添加它。然后，充电点应重新评估其收集的充电配置文件，以确定哪些充电配置文件将成为活动的。。

NOTE

不可能在没有活动事务存在的情况下或在事务之前将目的设置为TX配置文件的充电配置文件设置为TX配置文件。

NOTE

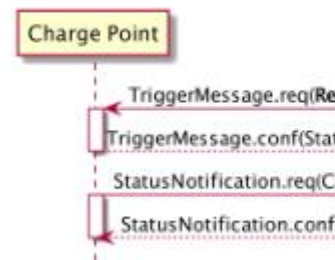
当在执行期间刷新充电配置文件时，建议将新的充电配置文件的开始时间表放在过去，因此在充电配置文件之间没有默认的充电行为。充电点应继续执行现有的充电配置文件，直到安装新的充电配置文件

NOTE

如果充电计划周期超过持续时间，则剩余周期将不会被执行。如果持续时间超过充电计划期，充电点应保持最后一个充电计划期的值，直到持续时间结束L

NOTE

当使用递归类型与低于递归类型周期的充电计划持续时间相结合时，充电点将在充电计划持续时间结束后恢复到默认行为



5.17. Trigger Message 触发信息

Figure 38. 序列图：触发消息

在正常运行期间，充电点将其状态和任何相关事件通知中央系统。如果没有任何报告，充电点将发送至少一个心脏跳动在预定的间隔。在正常情况下，这只是好的，但如果中央系统有（任何）理由怀疑最后一个已知的状态呢？如果一个中央系统正在进行固件更新，并且它收到的关于它的最后状态通知比合理预期的要长得多，它能做些什么？对于诊断请求的进度也可以这样要求。在这种情况下，问题不在于所需要的信息没有被现有的消息所覆盖，而在于时间问题。充电点有信息，但无法知道中央系统是否需要更新。

Trigger Message.req使中央系统能够请求充电点，发送由充电点发起的消息。在请求中，中央系统指示希望接收的消息。对于所有要求的信息，中央系统可能选择指示此请求适用于哪个连接器。请求的消息是引导的：如果指定的连接器ID与消息无关，则应该忽略它。在这种情况下，仍应发送所请求的电文。

相反，如果连接器ID是相关的，但没有，这应该解释为“对于所有允许的连接器ID值”。例如，对连接器ID0状态通知的请求是对充电点状态的请求。没有连接器ID的状态通知请求是对多个状态通知的请求：充电点本身的通知和每个连接器的通知。

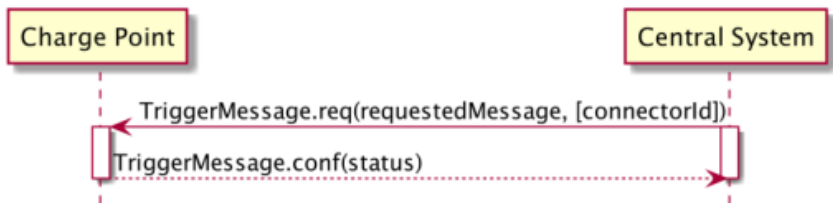


Figure 39. 序列图：触发消息状态通知示例

收费点应首先发送触发消息响应，然后发送请求的消息。在触发器消息.conf中，充电点应该通过返回接受或拒绝来指示它是否会发送它。如果它接受或拒绝发送的请求，则由收费点决定。如果请求的消息未知或未实现，充电点将返回NOT_IMPLEMENTED。

应发送被接受的充电点标记的消息。这种情况可能会发生，在接受请求和实际发送请求消息之间，同一消息由于正常操作而被发送。在这种情况下，刚刚发送的消息可能被视为符合请求

触发消息机制不打算检索历史数据。它触发的消息应该只提供当前信息。以这种方式触发的计量器值消息，例如应该返回配置键计数值采样数据中配置的所有计量器的最新测量值。启动事务和停止事务已经被排除在这个机制之外，因为它们不是状态相关的，而是根据它们的性质描述了一个转换。

5.18. Unlock Connector 解锁连接器

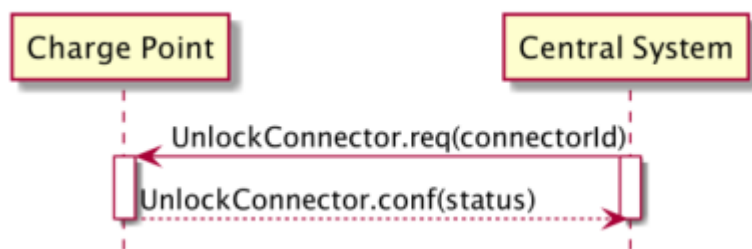


Figure 40. 顺序图：解锁连接器

中央系统可以请求充电点来解锁连接器。为此，充电点将发送一个解锁连接器.reqPDU。

此消息的目的：帮助电动汽车司机有问题，拔出他们的电缆从充电点，以防故障的连接器电缆保留。当EV驱动程序调用CPO服务台时，操作员可以手动触发将解锁连接器.req发送到充电点，从而迫使重新尝试解锁连接器。希望这次连接器解锁，EV驱动程序可以拔掉电缆并驱动。

Unlock Connector.req应该不用于远程停止正在运行的事务，而是使用远程停止事务。

收到解锁连接器.reqPDU后，充电点将用解锁连接器.confPDU响应。响应PDU应指示充电点是否能够解锁其连接器。

如果在特定的连接器上有一个正在进行的事务，那么充电点将首先完成事务，如停止事务中所描述的那样。

IMPORTANT

解锁连接器.req只用于解锁连接器上的电缆保持锁，而不是解锁连接器访问门。

5.19. Update Firmware 更新固件

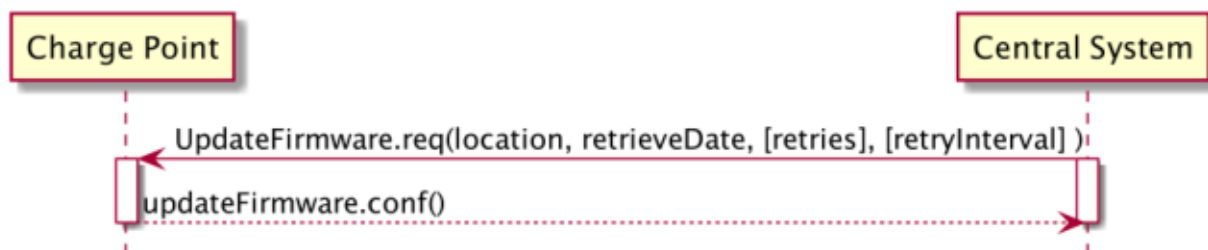


Figure 41. 顺序图：更新固件

中央系统可以通知充电点，它需要更新其固件。中央系统应该发送一个更新固件.reqPDU来指示充电点安装新的固件。PDU应包含一个日期和时间，在此之后，充电点被允许检索新固件和可以下载固件的位置。

在收到更新固件.reqPDU后，充电点将使用更新固件.confPDU响应。充电点应该在检索日期后尽快开始检索固件。

6. Messages 信息

6.1. Authorize.req 授权.req

这包含由充电点发送到中央系统的Authorize.reqPDU的字段定义。 另见授权

Field Name 字段名称	Field Type 字段类型	Card.	Description
idTag	IdToken	1..1	必须的。 这包含需要授权的标识符。

6.2. Authorize.conf 授权.conf

这包含由中央系统发送到充电点的Authorize.confPDU的字段定义， 以响应Authorize.reqPDU。 另见授权

Field Name	Field Type	Card.	Description
idTagInfo	IdTagInfo	1..1	必须的。 这包含有关授权状态、过期和父id的信息。

6.3. BootNotification.req 启动通知

这包含由充电点发送到中央系统的引导通知.reqPDU的字段定义。 另见引导通知

Field Name	Field Type	Card.	Description
充电盒序列号	CiString25Type	0..1	可选的。 这包含一个值该值标识充电点内的充电盒的序列号。 已折旧将在未来版本中删除
充电点模型	CiString20Type	1..1	必须的。 这包含一个值， 充电点。

Field Name	Field Type	Card.	Description
充电点序列号	CiString25Type	0..1	可选的。这包含一个值，该值标识充电点的序列号。
收费点供应商	CiString20Type	1..1	必须的。这包含一个值该值标识收费点的供应商。
固件版本	CiString50Type	0..1	可选的。这包含充电点的固件版本。
iccid	CiString20Type	0..1	可选的。这包含调制解调器SIM卡的ICCID。
imsi	CiString20Type	0..1	可选的。这包含调制解调器SIM卡的IMSI。
meterSerialNumber	CiString25Type	0..1	可选的。这包含充电点主功率计的序列号。
meterType	CiString25Type	0..1	可选的。这包含充电点的主功率计的类型。

6.4. BootNotification.conf

这包含由中央系统发送到充电点的Boot Notification.confPDU的字段定义，以响应Boot Notification.reqPDU
另见引导通知

Field Name	Field Type	Card.	Description
currentTime	dateTime	1..1	必须的。这包含了中央系统当前的时间。

Field Name	Field Type	Card.	Description
interval	integer	1..1	必须的。当注册状态被接受时，这包含以秒为单位的心跳间隔。如果中央系统返回的东西不是接受的则间隔字段的值指示发送下一个引导通知请求之前的最小等待时间
status	RegistrationStatus	1..1	必须的。这包含收费点是否已在系统中心注册。

6.5. CancelReservation.req 取消预订.req

这包含由中央系统发送到充电点的Cancel Reservation.reqPDU的字段定义。另见取消预订

Field Name	Field Type	Card.	Description
reservationId	integer	1..1	必须的。预订取消的ID.

6.6. CancelReservation.conf 取消预订.conf

这包含由充电点发送到中央系统的Cancel Reservation.confPDU的字段定义，以响应Cancel Reservation.reqPDU。另见取消预订

Field Name	Field Type	Card.	Description
status	CancelReservationStatus	1..1	必须的。这表明中央系统取消保留的成败.

6.7. ChangeAvailability.req 更改可用性.req

这包含由中央系统发送到充电点的更改可用性.reqPDU的字段定义。 另见更改可用性

Field Name	Field Type	Card.	Description
connectorId	integer connectorId >= 0	1..1	必须的。 可用性需要更改的连接器的id。 如果充电点及其所有连接器的可用性需要更改, 则使用Id'0'(零).
type	AvailabilityType	1..1	必须的。 这包含充电点应该执行的可用性更改类型.

6.8. ChangeAvailability.conf 更改可用性

这包含了ChangeAvability.confPDU返回到中央系统的字段定义。 另见更改可用性

Field Name	Field Type	Card.	Description
status	AvailabilityStatus	1..1	必须的。 这表明充电点是否能够执行可用性更改.

6.9. ChangeConfiguration.req 更改配置.req

这包含由中央系统发送到充电点的ChangeConfiguration.reqPDU的字段定义。 建议在收费点和中央系统之间商定“关键”和“价值”字段的内容和含义。 另见更改配置

Field Name	Field Type	Card.	Description
key	CiString50Type	1..1	必须的。要更改的配置设置的名称。参见标准配置键名和关联值
value	CiString500Type	1..1	必须的。作为设置的字符串的新值。参见标准配置键名和关联值

6.10. ChangeConfiguration.conf 更改配置.conf

这包含从充电点返回到中央系统的ChangeConfiguration.confPDU的字段定义。另见更改配置

Field Name	Field Type	Card.	Description
status	ConfigurationStatus	1..1	必须的。返回配置更改是否已被接受。

6.11. ClearCache.req

这包含由中央系统发送到充电点的ClearCache.reqPDU的字段定义。另见清除缓存

没有定义字段。

6.12. ClearCache.conf 清除缓存.conf

这包含由充电点发送到中央系统的ClearCache.confPDU的字段定义，以响应ClearCache.reqPDU。另见清除缓存

Field Name	Field Type	Card.	Description
status	ClearCacheStatus	1..1	必须的。如果收费点已执行请求，则接受，否则拒绝。

6.13. ClearChargingProfile.req 清晰的充电配置文件.req

这包含由中央系统发送到充电点的Clear充电配置文件.reqPDU的字段定义.

中央系统可以使用此消息清除（删除）特定的充电配置文件(由id表示)或与可选连接器ID、堆栈级别和充电配置文件目的字段的值匹配的充电配置文件的选择。 另见明确的充电配置文件

Field Name	Field Type	Card.	Description
id	integer	0..1	可选的。 充电配置文件的ID要清除.
connectorId	integer	0..1	可选的。 指定用于清除充电配置文件的连接器的ID。 连接器ID为零（0）指定整个充电点的充电配置文件。 缺少此参数意味着清除应用于与请求的其他标准匹配的所有充电配置文件。 .
chargingProfilePurpose	ChargingProfilePurposeType	0..1	可选的。 指定将被清除的充电配置文件的目的， 如果它们符合请求中的其他标准.
stackLevel	integer	0..1	可选的。 指定将清除充电配置文件的堆栈级别， 如果它们符合请求中的其他标准t

6.14. ClearChargingProfile.conf 清除充电配置文件.Conf

这包含由充电点发送到中央系统的ClearChargingProfile.confPDU的字段定义， 以响应ClearChargingProfile.reqPDU。 另见明确的充电配置文件

Field Name	Field Type	Card.	Description
status	ClearChargingProfileStatus	1..1	必须的。指示充电点是否能够执行请求。

6.15. DataTransfer.req 数据传输.req

这包含由中央系统发送到充电点的DataTransfer.reqPDU的字段定义，反之亦然。另见数据传输

Field Name	Field Type	Card.	Description
vendorId	CiString255Type	1..1	必须的。这确定了供应商的具体实现
messageId	CiString50Type	0..1	可选的。附加识别字段
data	Text Length undefined	0..1	可选的。没有指定长度或格式的数据。

6.16. DataTransfer.conf

这包含由充电点发送到中央系统的数据传输.confPDU的字段定义，反之亦然，DataTransfer.reqPDU。另见数据传输

Field Name	Field Type	Card.	Description
status	DataTransferStatus	1..1	必须的。这表明数据传输的成败。
data	Text Length undefined	0..1	可选的。应要求提供的数据。

6.17. DiagnosticsStatusNotification.req 诊断状态通知.req

这包含由充电点发送到中央系统的DiagnosticsStatusNotification.reqPDU的字段定义。另见诊断状态通知

Field Name	Field Type	Card.	Description
status	DiagnosticsStatus	1..1	必须的。这包含诊断上传的状态。

6.18. DiagnosticsStatusNotification.conf 诊断状态通知.conf

这包含由中央系统发送到充电点的DiagnosticsStatusNotification.confPDU的字段定义，以响应DiagnosticsStatusNotification.reqPDU。另见诊断状态通知

No fields are defined.

6.19. FirmwareStatusNotification.req 固件状态通知.req

这包含由充电点发送到中央系统的FirmwareStatusNotification.reqPDU的字段定义。请参阅固件状态通知

Field Name	Field Type	Card.	Description
status	FirmwareStatus	1..1	必须的。这包含固件安装的进度状态。

6.20. FirmwareStatusNotification.conf 固件状态通知.conf

这包含由中央系统发送到充电点的firmwareStatusNotification.confPDU的字段定义，以响应firmwareStatusNotification.reqPDU。请参阅固件状态通知

No fields are defined.

6.21. GetCompositeSchedule.req 得到综合时间表.req

这包含由中央系统发送到充电点的GetCompositeSchedule.reqPDU的字段定义。参见获取复合时间表

Field Name	Field Type	Card.	Description
connectorId	integer	1..1	必须的。请求时间表的连接器的ID。当连接器ID=0时，充电点将计算网格连接的预期消耗。
duration	integer	1..1	必须的。几秒钟的时间所要求的时间表的长度
chargingRateUnit	ChargingRateUnitType	0..1	可选的。可用于强制功率或电流剖面

6.22. GetCompositeSchedule.conf 得到综合时间表.conf

这包含充电点响应GetCompositeSchedule.req PDU发送到中央系统的GetCompositeSchedule.conf PDU的字段定义。参见获取复合时间表

Field Name	Field Type	Card.	Description
status	GetCompositeScheduleStatus	1..1	必须的。请求的状态。充电点将指示它是否能够处理请求
connectorId	integer	0..1	可选的。此通知中包含的收费时间表适用于连接器。
scheduleStart	DateTime	0..1	可选的。时间到了。充电配置文件中包含的周期相对于这个时间点。
chargingSchedule	ChargingSchedule	0..1	可选的。计划的复合充电时间表，能源消耗随着时间的推移。总是相对于时间表开始。

6.23. GetConfiguration.req 获取配置.req

这包含由中央系统发送到充电点的GetConfiguration.reqPDU的字段定义。 另见获取配置

Field Name	Field Type	Card.	Description
key	CiString50Type	0..*	可选的。 请求配置值的键列表

6.24. GetConfiguration.conf 获取配置.conf

这包含由充电点发送到中央系统的GetConfiguration.confPDU的字段定义，以响应GetConfiguration.req。
另见获取配置

Field Name	Field Type	Card.	Description
configurationKey	KeyValue	0..*	可选的。 请求或已知键的列表
unknownKey	CiString50Type	0..*	可选的。 未知的请求密钥

6.25. GetDiagnostics.req 获取诊断

这包含由中央系统发送到充电点的GetDiagnostics.reqPDU的字段定义。 另见“获取诊断

Field Name	Field Type	Card.	Description
location	anyURI	1..1	必须的。 这包含诊断文件应上传到的位置(目录).
retries	integer	0..1	可选的。 这指定了多少次充电点必须尝试上传诊断之前放弃。 如果该字段不存在, 则由充电点决定要重试多少次.

Field Name	Field Type	Card.	Description
retryInterval	integer	0..1	可选的。以秒为单位的间隔，在此之后可以尝试重试。如果此字段不存在，则由充电点决定在尝试之间等待多长时间。
startTime	dateTime	0..1	可选的。这包含要包含在诊断中的最古老日志信息的日期和时间。
stopTime	dateTime	0..1	可选的。这包含要包含在诊断中的最新日志信息的日期和时间。

6.26. GetDiagnostics.conf 获取诊断.conf

这包含由充电点发送到中央系统以响应GetDiagnostics.reqPDU的GetDiagnostics.confPDU的字段定义。另见“获取诊断”

Field Name	Field Type	Card.	Description
fileName	CiString255Type	0..1	可选的。这包含将上载的具有诊断信息的文件的名称。当没有诊断信息时，此字段不存在。

6.27. GetLocalListVersion.req

这包含由中央系统发送到充电点的GetLocalListVersion.reqPDU的字段定义。参见获取本地列表版本

No fields are defined.

6.28. GetLocalListVersion.conf 获取本地列表版本.conf

这包含收费点向中央系统发送的Get Local List Version.conf PDU响应Get Local List Version.req PDU的字段定义。参见获取本地列表版本

Field Name	Field Type	Card.	Description
listVersion	integer	1..1	必须的。这包含充电点中本地授权列表的当前版本号。

6.29. Heartbeat.req 心跳.req

这包含由充电点发送到中央系统的Heartbeat.reqPDU的字段定义。也见心跳

No fields are defined.

6.30. Heartbeat.conf 心跳.conf

这包含由中央系统发送到充电点的Heartbeat.confPDU的字段定义，以响应Heartbeat.reqPDU。也见心跳

Field Name	Field Type	Card.	Description
currentTime	dateTime	1..1	必须的。这包含中央系统的当前时间。

6.31. MeterValues.req 电表值.req

这包含由充电点发送到中央系统的MeterValues.reqPDU的字段定义。另见仪表值

Field Name	Field Type	Card.	Description
connectorId	integer connectorId >= 0	1..1	必须的。这包含一个数字(>0)，指定电荷点的连接器“0”(零)用于指定主功率计。

Field Name	Field Type	Card.	Description
transactionId	integer	0..1	可选的。这些仪表样本相关的事务。
meterValue	MeterValue	1..*	必须的。抽样的电表值加时间戳。

6.32. MeterValues.conf 电表值.conf

这包含由中央系统发送到充电点的MeterValues.confPDU的字段定义，以响应MeterValues.reqPDU。另见仪表值

No fields are defined.

6.33. RemoteStartTransaction.req 远程启动事务.req

这包含由中央系统发送到充电点的RemoteStartTransaction.reqPDU的字段定义。参见远程启动事务

Field Name	Field Type	Card.	Description
connectorId	integer	0..1	可选的。启动事务的连接器的数量。连接器ID应>0
idTag	IdToken	1..1	必须的。收费点必须使用的标识符来启动事务。
chargingProfile	ChargingProfile	0..1	可选的。充电配置文件将由充电点用于请求的事务。充电配置文件的目的是必须设置为TX配置文件

6.34. RemoteStartTransaction.conf 远程启动事务.conf

这包含从充电点发送到中央系统的RemoteStartTransaction.confPDU的字段定义。参见远程启动事务

Field Name	Field Type	Card.	Description
status	RemoteStartStopStatus	1..1	必须的。状态，指示收费点是否接受启动事务的请求。

6.35. RemoteStopTransaction.req 远程停止事务.req

这包含由中央系统发送到充电点的RemoteStopTransaction.reqPDU的字段定义。参见远程停止事务

Field Name	Field Type	Card.	Description
transactionId	integer	1..1	必须的。请求停止的事务的标识符。

6.36. RemoteStopTransaction.conf 远程停止事务.conf

这包含从充电点发送到中央系统的RemoteStopTransaction.confPDU的字段定义。参见远程停止事务

Field Name	Field Type	Card.	Description
status	RemoteStartStopStatus	1..1	必须的。状态，指示收费点是否接受停止事务的请求。

6.37. ReserveNow.req 立刻准备.req

这包含中央系统发送到充电点的ReserveNow.reqPDU的字段定义。也见立刻准备

Field Name	Field Type	Card.	Description
connectorId	integer connectorId >= 0	1..1	必须的。这包含要保留的连接器的id。值为0意味着保留不是针对特定连接器的。

Field Name	Field Type	Card.	Description
expiryDate	dateTime	1..1	必须的。这包含预订结束的日期和时间。
idTag	IdToken	1..1	必须的。充电点必须预留连接器的标识符。
parentIdTag	IdToken	0..1	可选的。父idTag。
reservationId	integer	1..1	必须的。此预订的唯一id。

6.38. ReserveNow.conf 立刻准备.conf

这包含由收费点发送到中央系统的ReserveNow.confPDU的字段定义，以响应ReserveNow.reqPDU。也见预备队现在

Field Name	Field Type	Card.	Description
status	ReservationStatus	1..1	必须的。这表明保留的成败。

6.39. Reset.req 复位.req

这包含由中央系统发送到充电点的Reset.reqPDU的字段定义。另见重置

Field Name	Field Type	Card.	Description
type	ResetType	1..1	必须的。这包含充电点应该执行的重置类型。

6.40. Reset.conf 复位.conf

这包含充电点响应Reset.reqPDU发送到中央系统的Reset.confPDU的字段定义。另见重置

Field Name	Field Type	Card.	Description
status	ResetStatus	1..1	必须的。这表明充电点是否能够执行重置.

6.41. SendLocalList.req 发送本地列表.req

这包含由中央系统发送到充电点的SendLocalList.reqPDU的字段定义.

如果没有（空）本地授权列表，并且更新类型已满，则从列表中删除所有标识。请求没有（空）本地授权列表的差异更新将对列表没有影响。本地授权列表中的所有id标记必须是唯一的，不允许重复值。另见发送本地列表

Field Name	Field Type	Card.	Description
listVersion	integer	1..1	必须的。如果是完整的更新这是完整列表的版本号。对于差异更新，它是应用更新后列表的版本号.
localAuthorizationList	AuthorizationData	0..*	可选的。在完全更新的情况下，这包含形成新的本地授权列表的值列表。在差异更新的情况下，它包含要应用于收费点中的本地授权列表的更改。最大数量的授权数据元素是可用的配置键：发送本地列表最大长度
updateType	UpdateType	1..1	必须的。这包含此请求的更新类型(完全或差异).

6.42. SendLocalList.conf 发送本地列表.conf

这包含由收费点响应Send Local List.req PDU发送到中央系统的Send Local List.conf PDU的字段定义
另见发送本地列表

Field Name	Field Type	Card.	Description
status	UpdateStatus	1..1	必须的。 这表明收费点是否成功地接收和应用了本地授权列表的更新.

6.43. SetChargingProfile.req 设置充电配置文件.req

这包含由中央系统发送到充电点的SetChargingProfile.reqPDU的字段定义.

中央系统使用此消息将充电配置文件发送到充电点。 另见设置充电配置文件

Field Name	Field Type	Card.	Description
connectorId	integer	1..1	必须的。 充电配置文件适用的连接器。 如果连接器ID=0， 则消息包含充电点的总体限制.
csChargingProfiles	ChargingProfile	1..1	必须的。 充电配置文件将设置在充电点.

6.44. SetChargingProfile.conf

这包含由充电点发送到中央系统的SetChargingProfile.confPDU的字段定义， 以响应SetChargingProfile.req PDU。 另见设置充电配置文件

Field Name	Field Type	Card.	Description
status	ChargingProfileStatus	1..1	必须的。返回充电点是否能够成功地处理消息。这并不能保证遵循的信件。收费点可能需要考虑其他限制因素.

6.45. StartTransaction.req 开始交易.req

本节包含由充电点发送到中央系统的StartTransaction.req PDU的字段定义。另见“开始交易

Field Name	Field Type	Card.	Description
connectorId	integer connectorId > 0	1..1	必须的。这将标识使用充电点的哪个连接器.
idTag	IdToken	1..1	必须的。这包含必须启动事务的标识符.
meterStart	integer	1..1	必须的。这包含事务开始时连接器的WH中的仪表值.
reservationId	integer	0..1	可选的。这包含由于此事务而终止的保留的id.
timestamp	dateTime	1..1	必须的。这包含事务启动的日期和时间.

6.46. StartTransaction.conf 开始交易.conf

这包含中央系统响应Start Transaction.req PDU发送到充电点的Start Transaction.conf PDU的字段定义
另见“开始交易

Field Name	Field Type	Card.	Description
idTagInfo	IdTagInfo	1..1	必须的。这包含有关授权状态、过期和父id的信息.
transactionId	integer	1..1	必须的。这包含中央系统提供的事务id.

6.47. StatusNotification.req 状态通知.req

这包含由收费点发送到中央系统的StatusNotification.reqPDU的字段定义。 另见状态通知

Field Name	Field Type	Card.	Description
connectorId	integer connectorId >= 0	1..1	必须的。报告状态的连接器的id。如果状态为充电点主控制器，则使用Id'0'(零).
errorCode	ChargePointErrorCode	1..1	必须的。这包含由充电点报告的错误代码.
info	CiString50Type	0..1	可选的。附加免费格式信息相关错误.
status	ChargePointStatus	1..1	必须的。这包含充电点的当前状态.
timestamp	dateTime	0..1	可选的。状态报告的时间。如果没有收到消息的时间将被假定.

Field Name	Field Type	Card.	Description
vendorId	CiString255Type	0..1	可选的。这确定了特定于供应商的实现。
vendorErrorCode	CiString50Type	0..1	可选的。这包含特定于供应商的错误代码。

6.48. StatusNotification.conf 状态通知.conf

这包含由中央系统响应StatusNotification.req PDU发送到充电点的StatusNotification.confPDU的字段定义。另见状态通

No fields are defined.

6.49. StopTransaction.req 停止交易

这包含由充电点发送到中央系统的StopTransaction.reqPDU的字段定义。另见停止交易

Field Name	Field Type	Card.	Description
idTag	IdToken	0..1	可选的。这包含请求停止充电的标识符。它是可选的，因为充电点可以在没有id标签的情况下终止充电，例如。以防重置。如果已知，充电点将发送id标签。
meterStop	integer	1..1	必须的。这包含事务结束时连接器的WH中的仪表值。
timestamp	dateTime	1..1	必须的。这包含事务停止的日期和时间。

Field Name	Field Type	Card.	Description
transactionId	integer	1..1	必须的。这包含Start Transaction.conf接收的事务ID.
reason	Reason	0..1	可选的。这包含了停止事务的原因。只有当“原因”为“局部”时，才可能省略”。
transactionData	MeterValue	0..*	可选的。这包含与计费目的相关的事务使用细节。

6.50. StopTransaction.conf

这包含由中央系统发送到充电点的StopTransaction.confPDU的字段定义，以响应StopTransaction.reqPDU。
另见停止交易

Field Name	Field Type	Card.	Description
idTagInfo	IdTagInfo	0..1	可选的。这包含有关授权状态过期和父id的信息。它是可选的，因为事务可能在没有标识符的情况下被停止。

6.51. TriggerMessage.req 触发信息.req

这包含由中央系统发送到充电点的触发器消息.reqPDU的字段定义。另见触发消息

Field Name	Field Type	Card.	Description
requestedMessage	MessageTrigger	1..1	必须的.
connectorId	integer connectorId > 0	0..1	可选的。只有当请求应用于特定连接器时才填写。

6.52. TriggerMessage.conf 触发信息.conf

这包含由充电点响应TriggerMessage.req PDU发送到中央系统的TriggerMessage.conf PDU的字段定义。 另见触发消息

Field Name	Field Type	Card.	Description
status	TriggerMessageStatus	1..1	必须的。 指示充电点是否会发送请求通知.

6.53. UnlockConnector.req 解锁连接器.req

这包含由中央系统发送到充电点的UnlockConnector.reqPDU的字段定义。 另见解锁连接器

Field Name	Field Type	Card.	Description
connectorId	integer connectorId > 0	1..1	必须的。 这包含要解锁的连接器的标识符.

6.54. UnlockConnector.conf

这包含由充电点发送到中央系统的UnlockConnector.confPDU的字段定义， 以响应UnlockConnector.reqPDU 另见解锁连接器

Field Name	Field Type	Card.	Description
status	UnlockStatus	1..1	必须的。 这表明充电点是否解锁了连接器.

6.55. UpdateFirmware.req 更新固件.req

这包含由中央系统发送到充电点的UpdateFirmware.reqPDU的字段定义。 请参阅更新固件

Field Name	Field Type	Card.	Description
location	anyURI	1..1	必须的。这包含一个字符串，其中包含指向要检索固件的位置的URI.
retries	integer	0..1	可选的。这指定了多少次充电点必须尝试下载固件之前放弃。如果该字段不存在，则由充电点决定要重试多少次.
retrieveDate	dateTime	1..1	必须的。这包含充电点必须检索（新）固件的日期和时间.
retryInterval	integer	0..1	可选的。以秒为单位的间隔，在此之后可以尝试重试。如果此字段不存在，则由充电点决定在尝试之间等待多长时间.

6.56. UpdateFirmware.conf 更新固件.conf

这包含由充电点发送到中央系统的UpdateFirmware.confPDU的字段定义，以响应UpdateFirmware.reqPDU。
请参阅更新固件

No fields are defined.

7. Types

7.1. AuthorizationData 授权数据

Class

构成本地授权列表更新条目的元素.

Field Name	Field Type	Card.	Description
idTag	IdToken	1..1	必须的。此授权适用的标识符.
idTagInfo	IdTagInfo	0..1	可选的。（当更新类型已满时需要）这包含有关授权状态过期和父id的信息。对于差分更新，以下应用程序：如果存在此元素，则应在本地授权列表中添加或更新此条目。如果没有此元素，则应删除本地授权列表中此idtag的条目.

7.2. AuthorizationStatus 授权状态

Enumeration

对Authorize.req的响应状态.

Value	Description
Accepted 接受	标识符允许充电.
Blocked 封锁了	标识符已被阻塞。不允许充电.
Expired 过期了	标识符已过期。不允许充电.
Invalid 无效	标识符未知。不允许充电.

Value	Description
ConcurrentTx 并发TX	标识符已经涉及到另一个事务，并且不允许多个事务。（仅与Start Transaction.req相关。）

7.3. AvailabilityStatus 可用性状态

Enumeration

响应Change Availability.req返回的状态。

Value	Description
Accepted 接受	请求已被接受并将被执行。
Rejected 拒绝了	请求尚未被接受，将不予执行。
Scheduled 预定的	请求已被接受，并将在进行中的事务完成后执行。

7.4. AvailabilityType 可用性类型

Enumeration

在Change Availability.req中请求的可用性更改。

Value	Description
Inoperative 不起作用	充电点不可用于充电。
Operative 执行部分	充电点可供充电。

7.5. CancelReservationStatus 取消预订状态

Enumeration

状态在Cancel Reservation.conf中。

Value	Description
Accepted 接受	标识符的保留已被取消。
Rejected 拒绝了	无法取消预订，因为标识符没有处于活动状态的预订。

7.6. ChargePointErrorCode 充电点错误代码

Enumeration

充电点状态上报在 [StatusNotification.req](#)中

Value	Description
ConnectorLockFailure 连接器锁定失败	未能锁定或解锁连接器.
EVCommunicationError 通信错误	与车辆的通信故障, 可能是模式3或其他通信协议问题. 这不是一个真正的错误, 因为充电点不需要进入故障状态. 相反, 它应该进入挂起的EVSE状态.
GroundFailure 地面故障	接地故障电路灭弧室已启动.
HighTemperature 高温	充电点内部温度太高.
InternalError 内部错误	内部硬或软件组件中的错误.
LocalListConflict 本地名单冲突	从中央系统收到的授权信息与本地授权列表冲突.
NoError 无错误	无报错.
OtherError 其他错误	其他类型的错误. 供应商错误代码中的更多信息.
OverCurrentFailure 过流故障	过流保护装置已跳闸.
OverVoltage 过压故障	电压已上升到可接受的水平以上.
PowerMeterFailure 功率表故障	未读取功率计. .
PowerSwitchFailure 电源开关故障	未能控制电源开关.
ReaderFailure	失败的id标签阅读器.
ResetFailure 复位失败	无法执行重置.
UnderVoltage 欠压故障	电压已降至可接受水平以下.
WeakSignal 信号弱	无线通信设备报弱信号.

7.7. ChargePointStatus 充电点状态

Enumeration

StatusNotification.req中报告的状态。可以为充电点主控制器(连接器ID=0)或特定连接器报告状态。充电点主控制器的状态是枚举的子集：可用、不可用或故障。

考虑操作的国家是：可用，准备，充电，暂停EVSE，暂停EV，整理，保留。被认为不起作用的国家是：无人可用，有过错。

Status	Condition
Available 可用	当连接器可用于新用户时（操作）
Preparing 准备工作	当连接器不再为新用户可用，但没有充电会话处于活动状态时。通常，当用户显示标签、插入电缆或车辆占用停车舱（操作）时，连接器被占用）
Charging 充电	当连接器的接触器关闭时，允许车辆充电（操作）
SuspendedEVSE 暂停EVSE	当连接器的接触器应EVSE的要求打开时，例如。由于智能充电限制或由于StartTransaction.conf的结果，指示不允许充电（操作）
SuspendedEV 暂停 EV	当EVSE准备提供能量，但接触器是打开的，例如。电动汽车还没有准备好。
Finishing 完成	当充电会话在连接器上停止时，但连接器尚未对新用户可用，例如。电缆尚未拆除或车辆未离开停车舱（操作）
Reserved 预留	当连接器由于“立即预备”命令（操作）而被保留时）

Status

Unavailable 不可用

Condition

当连接器因“更改可用性”命令或充电点自行决定转换为不可用的事件而不可用时。在收到“更改可用性”命令后，状态可能立即更改或更改可能被调度。当计划时，状态通知应在可用性更改生效（不生效）时发送

Faulted 故障

当充电点或连接器报告报告了一个错误，并且不能用于能量传递（不起作用）。

7.8. ChargingProfile 充电配置文件

Class

充电配置文件由一个充电时间表组成，描述每个时间间隔可以交付的功率或电流的数量。

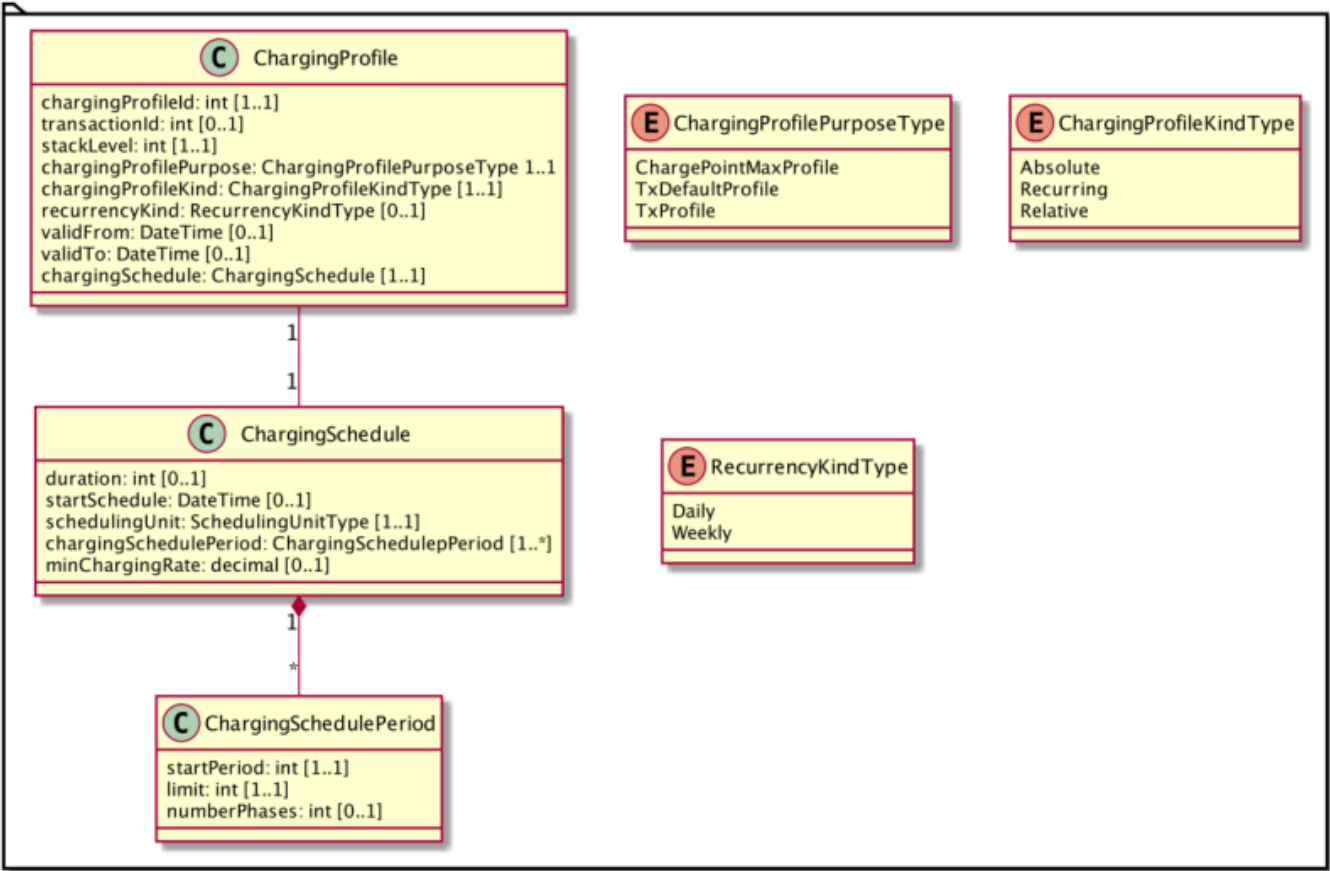


Figure 42. Class Diagram: 充电配置文件

Field Name	Field Type	Card.	Description
chargingProfileId 充电配置文件ID	integer	1..1	必须的。此配置文件的唯一标识符。
transactionId 交易ID	integer	0..1	可选的。只有当充电配置文件目的设置为TX配置文件时事务ID才能用于将配置文件与特定事务匹配。
stackLevel 堆栈级别	integer >=0	1..1	必须的。配置文件层次结构堆栈中的值确定级别。较高的值优先于较低的值。最低水平为0。
chargingProfilePurpose 充电配置文件的目的	ChargingProfilePurposeType	1..1	必须的。定义此消息传输的调度的目的。
chargingProfileKind 充电配置文件种类	ChargingProfileKindType	1..1	必须的。表示日程的种类。
recurrencyKind 递归类型	RecurrencyKindType	0..1	可选的。指示复发的起始点。
validFrom 有效来源	DateTime	0..1	可选的。配置文件开始有效的时间点。如果不存在，配置文件一旦被充电点接收就有效。当充电配置文件的目的为TX配置文件时，不要使用。

Field Name	Field Type	Card.	Description
validTo 有效的	DateTime	0..1	可选的。该配置文件停止有效的时间点。如果不存在，配置文件在被另一个配置文件替换之前是有效的。当充电配置文件的目的是TX配置文件时，不要使用。
chargingSchedule 收费时间表	ChargingSchedule	1..1	必须的。包含可用功率或电流随时间的限制。

7.9. ChargingProfileKindType

Enumeration

Value	Description
Absolute 绝对的	计划周期是相对于计划中定义的固定时间点。
Recurring 反复出现	时间表在第一个时间表期间定期重新启动。
Relative 相对的	时间表是相对于特定情况的起点（如会话的开始），这是由电荷点决定的。

7.10. ChargingProfilePurposeType

Enumeration

Value	Description
ChargePointMaxProfile 充电点最大配置文件	配置的最大功率或电流可用于整个充电点。设置充电配置文件.req消息。
TxDefaultProfile TX默认配置文件	用于新事务的默认配置文件。
TxProfile TX配置文件	配置文件与约束将由收费点对当前事务施加。当事务终止时具有此目的的配置文件将停止有效。

7.11. ChargingProfileStatus 充电配置文件状态

Enumeration

回复的状态在 [SetChargingProfile.req](#) 中.

Value	Description
Accepted 接受	请求已被接受并将被执行.
Rejected 拒绝	请求尚未被接受, 将不予执行.
NotSupported 并不支持	充电点表示不支持请求.

7.12. ChargingRateUnitType 充电率单元类型

Enumeration

定义充电计划的单元, 如: [GetCompositeSchedule.req](#)和充电计划

Value	Description
W	Watts (power).
A	Amperes (current).

7.13. ChargingSchedule

Class

Field Name	Field Type	Card.	Description
duration	integer	0..1	可选的。充电时间表的持续时间以秒为单位。如果持续时间为空, 则最后一个期间将无限期地继续下去, 或者在没有启动时间表的情况下, 直到事务结束.

Field Name	Field Type	Card.	Description
startSchedule	DateTime	0..1	可选的。绝对时间表的起点如果没有，时间表将相对于开始充电.
chargingRateUnit	ChargingRateUnitType	1..1	必须的。计量极限单位用.
chargingSchedulePeriod	ChargingSchedulePeriod	1..*	必须的。充电计划周期元素列表，定义最大功率或当前使用随着时间的推移.
minChargingRate	decimal	0..1	可选的。电动汽车支持的最小充电率。计量单位由费率单位定义。该参数用于本地智能充电算法，以优化功率分配，因为在充电过程在较低的充电速率下效率低下的情况下。最多接受一位数的分数(例如。8.1)

7.14. ChargingSchedulePeriod

Class

Field Name	Field Type	Card.	Description
startPeriod	integer	1..1	必须的。开始的时期，从时间表开始的秒。开始周期的值还定义了上一个周期的停止时间.

Field Name	Field Type	Card.	Description
limit	decimal	1..1	必须的。在计划期间的功率限制，用安培表示。最多接受一位数的分数(例如。 8.1)
numberPhases	integer	0..1	可选的。可用于充电的相数。如果需要多个阶段，除非给出另一个数字，否则将假定数字阶段=3.

7.15. CiString20Type

Class

通用的20个字符的不区分大小写字符串.

Field Name	Field Type	Description
cistring20	CiString[20]	字符串不区分大小写.

7.16. CiString25Type

Class

通用的25个字符的不区分大小写字符串.

Field Name	Field Type	Description
cistring25	CiString[25]	字符串不区分大小写.

7.17. CiString50Type

Class

通用的50个字符的不区分大小写字符串.

Field Name	Field Type	Description
ciString50	CiString[50]	字符串不区分大小写.

7.18. CiString255Type

Class

通用的255个字符的不区分大小写字符串.

Field Name	Field Type	Description
cistring255	CiString[255]	字符串不区分大小写.

7.19. CiString500Type

Class

通用的500个字符的不区分大小写字符串.

Field Name	Field Type	Description
ciString500	CiString[500]	字符串不区分大小写.

7.20. ClearCacheStatus 清除缓存状态

Enumeration

回复的状态在 [ClearCache.req](#)中.

Value	Description
Accepted 接受	命令已经执行.
Rejected 拒绝	命令尚未执行.

7.21. ClearChargingProfileStatus 清除充电配置文件状态

Enumeration

回复的状态在 [ClearChargingProfile.req](#)中.

Value	Description
Accepted	请求已被接受并将被执行.
Unknown	没有发现与请求匹配的充电配置文件.

7.22. ConfigurationStatus 配置状态

Enumeration

状态在 [ChangeConfiguration.conf](#)中.

Value	Description
Accepted 接受	配置键支持和设置已更改.
Rejected 拒绝	配置键支持, 但设置无法更改.
RebootRequired 需要重新启动	支持配置键, 设置已更改, 但重新启动后将可用更改 (充电点不会重新启动本身)
NotSupported 不支持	不支持配置键.

7.23. DataTransferStatus 数据传输状态

Enumeration

状态在 [DataTransfer.conf](#)中.

Value	Description
Accepted 接受	消息已被接受, 所包含的请求被接受.
Rejected 拒绝了	消息已被接受, 但包含的请求被拒绝.
UnknownMessageId 未知消息ID	由于消息ID字符串未知, 无法解释消息.
UnknownVendorId 未知供应商ID	由于未知的供应商ID字符串, 无法解释消息.

7.24. DiagnosticsStatus 诊断状态

Enumeration

状态在 [DiagnosticsStatusNotification.req](#) 中.

Value	Description
Idle	充电点没有执行诊断相关任务。状态空闲仅用于由TriggerMessage.req触发的DiagnosticsStatusNotification.req中
Uploaded 已上传	诊断信息已上传。
UploadFailed 上传失败	上传诊断失败。
Uploading 正在上传	文件正在上传。

7.25. FirmwareStatus 固件状态

Enumeration

固件下载的状态，如报告中所示在 [FirmwareStatusNotification.req](#)中。

Value	Description
Downloaded	新固件已由充电点下载。
DownloadFailed	充电点未能下载固件。
Downloading	固件正在下载中。
Idle	充电点没有执行固件更新相关任务。状态空闲只能由TriggerMessage.req消息触发的FirmwareStatusNotification.req中使用
InstallationFailed	安装新固件失败。
Installing	正在安装固件。
Installed	新固件已成功安装在充电点。

7.26. GetCompositeScheduleStatus 获得复合时间表状态

Enumeration

回复的状态在 [GetCompositeSchedule.req](#)中。

Value	Description
Accepted	请求已被接受并将被执行。

Value	Description
Rejected	请求尚未被接受，将不予执行。

7.27. IdTagInfo

Class

包含有关标识符的状态信息。它在授权、开始事务和停止事务响应中返回。

如果没有给出到期日，则状态没有结束日期。

Field Name	Field Type	Description
expiryDate	dateTime	可选的。这包含了从授权缓存中删除id标记的日期。
parentIdTag	IdToken	可选的。这包含父标识符。
status	AuthorizationStatus	必须的。这包含id标签是否已被中央系统接受。

7.28. IdToken

Class

包含用于授权的标识符。它是一个不区分大小写的字符串。在未来的版本中，这可能成为支持多种形式标识符的复杂类型。

Field Name	Field Type	Description
IdToken	String[20]	必须的。ID令牌不区分大小写。

7.29. KeyValue

Class

包含有关特定配置键的信息。它在GetConfiguration.conf中返回。

Name	Field Type	Card.	Description
key	CiString50Type	1..1	必须的.
readonly	Boolean	1..1	必须的。 如果值可以用“更改配置”消息设置，则为false.
value	CiString500Type	0..1	可选的。 如果键已知但未设置， 则此字段可能不存在.

7.30. Location

Enumeration

采样值中值元素的可选“位置”字段的允许值.

Value	Description
Body	充电点内部测量(例如。 温度)
Cable	测量从电缆之间的EV和充电点
EV	电动汽车的测量
Inlet	测量在网络（“网格”）入口连接
Outlet	连接器上的测量。 默认值

7.31. Measurand

Enumeration

值元素的可选“度量”字段的允许值，如Meter Values.req和Stop Transaction.req消息中使用的。“度量”的默认值总是“Energy.Active.Import。 登记”

Value	Description
Current.Export	瞬时电流来自EV
Current.Import	瞬时电流流向EV
Current.Offered	提供给EV的最大电流
Energy.Active.Export.Register	由EV(Wh或kWH)输出的能量)
Energy.Active.Import.Register	由EV(Wh或kWH)输出的能量)

Value	Description
Energy.Reactive.Export.Register	由EV(varh或kvarh)输出的活性能
Energy.Reactive.Import.Register	由EV(varh或kvarh)进口的活性能
Energy.Active.Export.Interval	由EV(varh或kvarh)进口的活性能
Energy.Active.Import.Interval	由EV(Wh或kWh)进口的能量
Energy.Reactive.Export.Interval	电动汽车出口的反应能源。(varh或kvarh)
Energy.Reactive.Import.Interval	电动汽车进口的活性能。(varh或kvarh)
Frequency	瞬时读取电源线频率
Power.Active.Export	由EV输出的瞬时有功功率。(W或kW)
Power.Active.Import	由EV进口的瞬时有功功率。(W或kW)
Power.Factor	全能流瞬时功率因数
Power.Offered	提供给EV的最大功率
Power.Reactive.Export	由EV输出的瞬时无功。(var或kvar)
Power.Reactive.Import	由EV进口的瞬时无功。(var或kvar)
RPM	风扇转速在RPM
SoC	充电车辆充电状态百分比
Temperature	充电点内部温度读数.
Voltage	瞬时交流RMS电源电压

7.32. MessageTrigger 消息触发

Enumeration

在Trigger Message.req中触发的请求类型.

Value	Description
BootNotification	以触发启动通知请求
DiagnosticsStatusNotification	以触发诊断状态通知请求
FirmwareStatusNotification	触发固件状态通知请求
Heartbeat	触发心跳请求

Value	Description
MeterValues	以触发MeterValue请求
StatusNotification	以触发 StatusNotification 请求

7.33. MeterValue

Class

在Meter Value.req中收集一个或多个采样值。表值中的所有采样值都在同一时间点采样。

Field Name	Field Type	Card.	Description
Timestamp 时间戳	dateTime	1..1	必须的。测量值的时间戳(s)。
sampledValue 采样价值 SampledValue		1..*	必须的。一个或多个测量值

7.34. Phase

Enumeration

在采样值中使用的相位。相位指定如何解释测量值。请注意，并不是所有的相位值都适用于所有的测量。

Value	Description
L1	Measured on L1
L2	Measured on L2
L3	Measured on L3
N	Measured on Neutral
L1-N	Measured on L1 with respect to Neutral conductor
L2-N	Measured on L2 with respect to Neutral conductor
L3-N	Measured on L3 with respect to Neutral conductor
L1-L2	Measured between L1 and L2
L2-L3	Measured between L2 and L3
L3-L1	Measured between L3 and L1

7.35. ReadingContext

Enumeration

Values of the context field of a value in [SampledValue](#).

Value	Description
Interruption.Begin	中断开始时的值.
Interruption.End	中断后恢复时的值.
Other	任何其他情况的值.
Sample.Clock	在时钟对齐间隔时所取的值.
Sample.Periodic	作为相对于事务开始时间的周期样本的值.
Transaction.Begin	交易结束时的值.
Transaction.End	交易开始时的值.
Trigger	响应TriggerMessage.req的值

7.36. Reason

Enumeration

StopTransaction.req中事务的原因.

Value	Description
EmergencyStop	使用了紧急停止按钮.
EVDisconnected	断开电缆, 车辆远离感应充电单元.
HardReset	收到硬复位命令.
Local	应用户在充电点的请求在本地停止。这是交易的定期终止。 举例：出示RFID标签，按下按钮停止.
Other	任何其他原因.
PowerLoss	完全丧失电力.
Reboot	发生了本地启动的重置/重新启动。（例如看门狗踢进来）

Value	Description
Remote	应用户请求远程停止。 这是交易的定期终止。 例如使用智能手机应用程序终止，超过（非本地）预付信贷。
SoftReset	收到软复位命令。
UnlockCommand	中央系统发送了一个解锁连接器命令。
DeAuthorized	由于Start Transaction.conf中的授权状态，事务被停止

7.37. RecurrencyKindType

Enumeration

Value	Description
Daily	日程安排在第二天开始时重新开始。
Weekly	日程安排在下周初重新开始（定义为星期一上午）

7.38. RegistrationStatus

Enumeration

响应Boot Notification.req的注册结果。

Value	Description
Accepted	充电点由中央系统接受。
Pending	中央系统尚未准备好接受收费点。中央系统可以发送消息来检索信息或准备收费点。
Rejected	充电点不被中央系统接受。这可能发生在充电点id不知道的中央系统。

7.39. RemoteStartStopStatus 远程启动停止状态

Enumeration

RemoteStartTransaction.req或RemoteStopTransaction.req请求的结果.

Value	Description
Accepted	命令将被执行.
Rejected	命令将不会被执行.

7.40. ReservationStatus

Enumeration

状态在 [ReserveNow.conf](#) 中.

Value	Description
Accepted	已经预订了.
Faulted	未进行预留, 因为连接器或指定连接器处于故障状态.
Occupied	尚未预订。 所有连接器或指定连接器被占用.
Rejected	尚未预订。 收费点没有配置为接受预订.
Unavailable	未进行预订, 因为连接器或指定连接器处于不可用状态.

7.41. ResetStatus

Enumeration

Result of [Reset.req](#).

Value	Description
Accepted	命令将被执行。
Rejected	命令将不会被执行.

7.42. ResetType

Enumeration

Type of reset requested by [Reset.req](#).

Value	Description
Hard	完全重新启动充电点软件.
Soft	返回初始状态，优雅地终止正在进行的任何事务.

7.43. SampledValue

Class

电表中的单个采样值。每个值都可以附带可选字段.

Field Name	Field Type	Card.	Description
value	String	1..1	必须的。 值为“原始”（十进制数字或“签名数据”。 字段类型是“字符串”，允许数字签名的数据读数。 小数数值也是可以接受的，以允许分数值的测量如温度和电流.
context	ReadingContext	0..1	可选的。 详细值的类型： 开始，结束或样本默认=“示例。 周期性”
format	ValueFormat	0..1	可选的。 原始数据或签名数据。 默认=“Raw”
measurand	Measurand	0..1	可选的。 测量类型。 默认=“Energy.Active.Import。 登记”

Field Name	Field Type	Card.	Description
phase	Phase	0..1	可选的。 指示如何解释测量值 例如，在L1和中性(L1-N)之间 请注意，并不是所有的相位值 都适用于所有的测量。当没有 相位时，测量值被解释为一个 整体值。
location	Location	0..1	可选的。 测量的位置。 默认=“出口”
unit	UnitOfMeasure	0..1	可选的。 值的单位。 默认 =“Wh”，如果（默认）度量是 “能量”类型。

7.44. TriggerMessageStatus 触发消息状态

Enumeration

状态在 [TriggerMessage.conf](#) 中.

Value	Description
Accepted	请求的通知将被发送.
Rejected	请求的通知将不会发送.
NotImplemented	请求的通知不能发送，因为它要么没有实现，要么未知.

7.45. UnitOfMeasure 计量单位

Enumeration

值元素的可选“单元”字段的允许值，如Meter Values.req和Stop Transaction.req消息中使用的。“单位”的默认值总是“Wh”。

Value	Description
Wh	Watt-hours (energy). Default.
kWh	kiloWatt-hours (energy).
varh	Var-hours (reactive energy).
kvarh	kilovar-hours (reactive energy).
W	Watts (power).
kW	kilowatts (power).
VA	VoltAmpere (apparent power).
kVA	kiloVolt Ampere (apparent power).
var	Vars (reactive power).
kvar	kilovars (reactive power).
A	Amperes (current).
V	Voltage (r.m.s. AC).
Celsius	Degrees (temperature).
Fahrenheit	Degrees (temperature).
K	Degrees Kelvin (temperature).
Percent	Percentage.

7.46. UnlockStatus 解锁状态

Enumeration

响应UnlockConnector.req的状态.

Value	Description
Unlocked	连接器已成功解锁.
UnlockFailed	未能解锁连接器.
NotSupported	充电点没有连接器锁.

7.47. UpdateStatus 更新状态

Enumeration

SendLocalList.req的更新类型.

Value	Description
Accepted	本地授权列表成功更新.
Failed	未能更新本地授权列表.
NotSupported	更新本地授权列表不支持收费点.
VersionMismatch	差异更新请求中的版本号小于或等于当前列表的版本号.

7.48. UpdateType 更新类型

Enumeration

SendLocalList.req的更新类型.

Value	Description
Differential	指示必须用此消息中的值更新当前本地授权列表.
Full	指示必须将当前本地授权列表替换为此消息中的值。

7.49. ValueFormat

Enumeration

指定如何解释采样值中的值元素的格式.

Value	Description
Raw	数据将被解释为整数/小数数字数据.
SignedData	数据表示为有符号二进制数据块， 编码为十六进制数据.

8. Firmware and Diagnostics File Transfer

固件和诊断文件传输

This section is normative.

支持的传输协议由配置密钥支持的文件传输协议控制。ftp, ftps, http, https(csl)

8.1. Download Firmware

当一个充电点被通知新的固件，它需要能够下载这个固件。中央系统在请求中提供一个可以下载固件的URL。URL还包含必须用于下载固件的协议。

建议通过FTP或FTPS下载固件。FTP(S)比HTTP更适合于大二进制数据。此外，FTP(S)具有恢复下载的能力。如果下载中断，充电点可以在它已经下载的部分之后恢复下载。FTP URL的格式为：ftp: /user: password@host: port/path, 其中部件用户: password@、: password或: port可能被排除在外。

为了确保正确的固件被下载，建议固件也是数字签名。

8.2. Upload Diagnostics

当要求充电点上传诊断文件时，中央系统在请求中提供一个URL，其中充电点应该上传该文件。URL还包含必须用于上传文件的协议。

建议通过FTP或FTPS上传诊断文件。FTP(S)比HTTP更适合于大二进制数据。此外，FTP(S)具有恢复上传的能力。如果上传中断，充电点可以在它已经上传的部分之后恢复上传。FTPURL的格式为：ftp: /用户: 密码@主机: 端口/路径, 其中部件用户: 密码@、: 密码或: 端口可能被排除在外。

9. Standard Configuration Key Names & Values

标准配置键名和值

下面是在本规范中标准化的所有配置键的列表。该列表由Feature Profiles分隔。在特定配置文件下提到的所需配置密钥只有在支持该配置文件的情况下才需要由充电点支持。

对于具有布尔类型的可选配置密钥，以下规则适用于对GetConfiguration.req的响应中的配置密钥，而不包含密钥列表：

- 如果存在密钥，则充电点提供由密钥配置的功能，并且可以通过设置密钥的值来启用或禁用它。
- 如果密钥不存在，则充电点不提供可以由密钥配置的功能。

“可访问性”属性显示某个配置键的值是只读(“R”)还是读写(“RW”)。如果密钥是只读的，中央系统可以使用GetConfiguration读取密钥的值，但不能写入它。如果可访问性是读写的，中央系统也可以使用Change Configuration为密钥编写值。

9.1. Core Profile

9.1.1.AllowOfflineTxForUnknownId 允许离线TX为未知Id

Required/optional	optional
Accessibility	RW
Type	boolean
Description	如果此密钥存在，则收费点支持未知离线授权。如果此键报告true值，则启用未知离线授权。

9.1.2.AuthorizationCacheEnabled 授权缓存启用

Required/optional	optional
Accessibility	RW
Type	boolean
Description	如果存在此密钥，则收费点支持授权缓存。如果此键报告的值为true，则启用授权缓存。

9.1.3.AuthorizeRemoteTxRequests 授权远程TX请求

Required/optional	required
Accessibility	R or RW. Choice is up to Charge Point implementation.
Type	boolean
Description	远程请求以远程启动事务的形式启动事务。req消息是否应该像启动事务的本地操作一样事先被授权.

9.1.4.BlinkRepeat

Required/optional	optional
Accessibility	RW
Type	int
Unit	times
Description	信号时闪烁充电点照明的次数

9.1.5.ClockAlignedDataInterval 时钟对齐数据间隔

Required/optional	required
Accessibility	RW
Type	int
Unit	seconds
Description	时钟对齐数据间隔的大小(以秒为单位。这是每天零时00分（午夜)开始的一组均匀间隔的聚集间隔的大小(以秒为单位)。例如，值900（15分钟）表示每天应分为96个15分钟间隔。当时钟对齐数据被传输时，所涉及的间隔由开始时间和（可选）持续时间间隔值标识，根据ISO8601标准表示。所有“每期”数据(例如。能量读数应在整个间隔（或部分间隔，在充电会话的开始或结束时)累积(对于“流动”类型度量，如能量)，或平均(对于其他值)，并在每个间隔的末尾发送(如果启用)，并带有间隔开始时间时间戳。按照惯例，“0”（数字为零）的值将被解释为不应传输时钟对齐的数据.

9.1.6.ConnectionTimeOut 连接超时

Required/optional	required
Accessibility	RW
Type	int
Unit	seconds
Description	间隔（从成功授权），直到初始充电会话自动取消，因为EV用户未能（正确）插入充电电缆连接器到适当的连接器）。

9.1.7.GetConfigurationMaxKeys

Required/optional	required
Accessibility	R
Type	int
Description	Get Configuration.req PDU中请求的最大配置键数.

9.1.8.HeartbeatInterval

Required/optional	required
Accessibility	RW
Type	int
Unit	seconds
Description	与中央系统的不活动间隔(没有OCPP交换)，然后充电点应该发送Heartbeat.reqPDU

9.1.9.LightIntensity

Required/optional	optional
Accessibility	RW
Type	int
Unit	%
Description	照亮充电点照明的最大强度百分比

9.1.10.LocalAuthorizeOffline 本地授权离线

Required/optional	required
Accessibility	RW
Type	boolean
Description	当脱机时，收费点是否将启动本地授权标识符的事务。

9.1.11.LocalPreAuthorize 本地预授权

Required/optional	required
Accessibility	RW
Type	boolean
Description	收费点在联机时是否将启动本地授权标识符的事务，而无需等待或请求中央系统的Authorize.conf

9.1.12.MaxEnergyOnInvalidId 无效ID上的最大能量

Required/optional	optional
Accessibility	RW
Type	integer
Unit	Wh
Description	当一个标识符在事务开始后被中央系统失效时，以Wh表示的最大能量。

9.1.13.MeterValuesAlignedData 仪表值对齐数据

Required/optional	required
Accessibility	RW
Type	CSL
Description	时钟对齐的测量和(S)将包括在一个表值。reqPDU，每个时钟对齐的数据间隔秒

9.1.14.MeterValuesAlignedDataMaxLength 仪表值对齐数据最大长度

Required/optional	optional
-------------------	----------

Accessibility	R
Description	表值对齐数据配置键中的最大项数.

9.1.15.MeterValuesSampledData 仪表值采样数据

Required/optional	required
Accessibility	RW
Type	CSL
Description	采样测量包含在一个米值。reqPDU，每个米值样本间隔秒。在适用的情况下，Measurand与可选相位相结合；例如：电压。L1默认值：“Energy.Active.Import。登记”

9.1.16.MeterValuesSampledDataMaxLength 仪表值采样数据最大长度

Required/optional	optional
Accessibility	R
Description	表值采样数据配置键中的最大项数.

9.1.17.MeterValueSampleInterval 仪表值样本间隔

Required/optional	required
Accessibility	RW
Type	int
Unit	seconds
Description	计量（或其他）数据采样之间的间隔，打算通过“仪表值”PDU传输。对于充电会话数据(Connector Id>0)，从充电事务开始，在此间隔定期获取和传输样本。按照惯例，“0”（数字为零）的值应解释为不应传输采样数据.

9.1.18.MinimumStatusDuration 最低状态持续时间

Required/optional	optional
Accessibility	RW
Type	int

Unit	seconds
Description	在状态通知.reqPDU发送到中央系统之前，充电点或连接器状态稳定的最小持续时间.

9.1.19.NumberOfConnectors 连接器的数量

Required/optional	required
Accessibility	R
Type	int
Description	此充电点的物理充电连接器的数量.

9.1.20.ResetRetries 重置重试

Required/optional	required
Accessibility	RW
Type	int
Unit	times
Description	重试充电点未成功重置的次数.

9.1.21.ConnectorPhaseRotation 连接器相位旋转

Required/optional	required
Accessibility	RW
Type	CSL

Description	<p>每个连接器相对于连接器的能量计（或如果没有，网格连接）的相位旋转）。每个连接器可能的值是：不适用（对于单相或直流充电点）未知(尚未)已知)</p> <p>RST (标准参考阶段) RTS (反向参考相位) SRT (反转240度旋转) STR (标准120度旋转) TRS (标准240度旋转) TSR (反转120度旋转)</p> <p>可以将R识别为第一阶段(L1)，S识别为第二阶段(L2)，T识别为第三阶段(L3)。如果已知，充电点也可能使用索引号零 (0)报告网格连接与主能量计之间的相位旋转)。</p> <p>值以CSL报告，格式为： 0.RST, 1.RST, 2.RTS</p>
--------------------	---

9.1.22.ConnectorPhaseRotationMaxLength 连接器相位旋转最大长度

Required/optional	optional
Accessibility	R
Description	连接器相位旋转配置键中的最大项数.

9.1.23.StopTransactionOnEVSideDisconnect 停止交易在EVSide断开

Required/optional	required
Accessibility	RW
Type	boolean
Description	当设置为true时，当电缆从EV上拔出时，充电点将在管理上停止事务.

9.1.24.StopTransactionOnInvalidId 在无效ID上停止事务

Required/optional	required
Accessibility	RW
Type	boolean
Description	当收费点在StartTransaction.conf中接收到此事务的非接受授权状态时，是否会停止正在进行的事务

9.1.25.StopTxnAlignedData 停止TXN对齐数据

Required/optional	required
Accessibility	RW
Type	CSL
Description	时钟对齐的周期度量(S)将包含在StopTransaction.reqMeter值.reqPDU的事务数据元素中，用于充电会话的每个时钟对齐数据间隔

9.1.26.StopTxnAlignedDataMaxLength 停止TXN对齐数据最大长度

Required/optional	optional
Accessibility	R
Description	停止TXN对齐数据配置键中的最大项数.

9.1.27.StopTxnSampledData 停止TXN采样数据

Required/optional	required
Accessibility	RW
Type	CSL
Description	采样度量将包含在StopTransaction.reqPDU的事务数据元素中，从充电会话开始的每电表值示例间隔秒

9.1.28.StopTxnSampledDataMaxLength 停止TXN采样数据最大长度

Required/optional	optional
Accessibility	R
Description	停止Txn采样数据配置键中的最大项数.

9.1.29.SupportedFeatureProfiles 支持的特性概要

Required/optional	required
Accessibility	R
Type	CSL

Description	支持的功能概要列表。可能的配置文件标识符：核心，固件管理，本地自动列表管理预订，智能充电和远程触发。
--------------------	--

9.1.30.SupportedFeatureProfilesMaxLength 支持的特征轮廓最大长度

Required/optional	optional
Accessibility	R
Description	支持的特征配置文件中的最大项数 配置键。

9.1.31.TransactionMessageAttempts 事务消息尝试

Required/optional	required
Accessibility	RW
Type	int
Unit	times
Description	当中央系统无法处理事务时，收费点应该多久提交一次与事务相关的消息。

9.1.32.TransactionMessageRetryInterval 事务消息重试间隔

Required/optional	required
Accessibility	RW
Type	int
Unit	seconds
Description	在重新提交中央系统未能处理的与交易相关的消息之前，收费点应该等待多长时间。

9.1.33.UnlockConnectorOnEVSideDisconnect 解锁连接器在EVSide断开

Required/optional	required
Accessibility	RW
Type	boolean
Description	当设置为true时，充电点将在充电点一侧解锁电缆，当电缆在EV上被拔出时。

9.1.34.WebSocketPingInterval

Required/optional	optional
Accessibility	RW
Type	int
Unit	seconds
Description	仅与websocket实现相关。0禁用客户端websocket Ping/Pong。在这种情况下，要么没有乒乓球/乒乓球，要么服务器启动乒乓球，客户端用乒乓球响应。正值被解释为pings之间的秒数。负值是不允许的。更改配置将返回拒绝的结果。

9.2. Local Auth List Management Profile

本地自动列表管理配置文件

9.2.1.LocalAuthListEnabled 本地自动列表已启用

Required/optional	required
Accessibility	RW
Type	boolean
Description	是否启用本地授权列表

9.2.2.LocalAuthListMaxLength 本地自动列表最大长度

Required/optional	required
Accessibility	R
Type	int
Description	可存储在本地授权列表中的最大标识数

9.2.3.SendLocalListMaxLength 发送本地列表最大长度

Required/optional	required
Accessibility	R
Type	int
Description	可以在单个Send Local List.req中发送的标识的最大数量

9.3. Reservation Profile 保留简介

9.3.1. ReserveConnectorZeroSupported 备用连接器零支持

Required/optional	optional
Accessibility	R
Type	boolean
Description	如果此配置键存在并设置为true：充电点支持对连接器0的保留。

9.4. Smart Charging Profile 智能充电配置文件

9.4.1. ChargeProfileMaxStackLevel 充电配置文件最大堆叠水平

Required/optional	required
Accessibility	R
Type	int
Description	充电配置文件的最大堆栈级别。定义的数字还表示每个充电配置文件用途的安装充电计划的最大允许数量。

9.4.2. ChargingScheduleAllowedChargingRateUnit 充电时间表允许充电率单位

Required/optional	required
Accessibility	R
Type	CSL
Description	一份支持的数量清单，用于充电时间表。允许的值：“电流”和“功率”

9.4.3. ChargingScheduleMaxPeriods

Required/optional	required
Accessibility	R
Type	int
Description	每个充电计划可定义的最大周期数。

9.4.4.ConnectorSwitch3to1PhaseSupported 连接器开关3到1阶段支持

Required/optional	optional
Accessibility	R
Type	bool
Description	如果定义为true，则此充电点支持在充电会话期间从3阶段切换到1阶段.

9.4.5.MaxChargingProfilesInstalled 安装最大充电配置文件

Required/optional	required
Accessibility	R
Type	int
Description	一次安装的最大充电配置文件数

Appendix A: New in OCPP 1.6

The following changes are made in OCPP 1.6 compared to OCPP 1.5 [\[OCPP1.5\]](#):

- Smart Charging is added
- A binding to JSON over WebSocket as a transport protocol is added, reducing data usage and enabling OCPP communication through NAT routers, see: [OCPP JSON Specification](#)
- Extra statuses are added to the [ChargePointStatus](#) enumeration, giving the CPO and ultimately end-users more information about the current status of a Charge Point
- Structure of MeterValues.req is changed to eliminate use of XML Attributes, this is needed for support of JSON (no attribute support in JSON).
- Extra values are added to the [Measurand](#) enumeration, giving Charge Point manufacturers the possibility to send new information to a Central System, such as the State of Charge of an EV
- The [TriggerMessage](#) message is added, giving the Central System the possibility to request information from the Charge Point
- A new *Pending* member is added to the [RegistrationStatus](#) enumeration used in [BootNotification.conf](#)
- More and clearer configuration keys are added, making it clearer to the CPO how to configure the different business cases in a Charge Point
- The messages and configuration keys are split into profiles, making it easier to implement OCPP gradually or only in part
- Known ambiguities are removed (e.g. when to use [UnlockConnector.req](#), how to respond to [RemoteStart/Stop](#), Connector numbering)

A.1. Updated/New Messages:

- [BootNotification.req](#)
 - Change IccId and Imsi to CiString[] to enforce maximum lengths.
- [BootNotification.conf](#)
 - heartbeatInterval to interval, interval now also used for other purposes than heartbeat, need to fix in spec
 - Added status Pending
- [ChargePointErrorCode](#)
 - Added enum values: [InternalError](#), [LocalListConflict](#) and [UnderVoltage](#)
 - Renamed enum value [Mode3Error](#) to [EVCommunicationError](#)
- [ChargePointStatus](#)
 - Replaced enum value [Occupied](#) with the more detailed values: [Preparing](#), [Charging](#),

SuspendedEVSE,SuspendedEV andFinishing

- [ChargingRateUnitType](#)
 - New
- [ConfigurationStatus](#)
 - Added enumRebootRequired
- [ClearChargingProfile.req](#)
 - New
- [ClearChargingProfile.conf](#)
 - New
- [DiagnosticsStatus](#)
 - Added enumUploading andIdle
- [FirmwareStatus](#)
 - Added enumDownloading,Installing andIdle
- [GetCompositeSchedule.req](#)
 - New
- [GetCompositeSchedule.conf](#)
 - New
- [Location](#)
 - Added enumCable andEV
- [Measurand](#)
 - Added enumCurrent.Offered,Frequency,Power.Offered,RPM andSoC
- [MeterValues.req](#)
 - overhaul of complex data structures
 - Added 'phase' field
- [ReadingContext](#)
 - Added enumTrigger andOther
- [RemoteStartTransaction.req](#)
 - Added [ChargingProfile](#) optional
- [SendLocalList.req](#)
 - removed hash
- [SendLocalList.conf](#)
 - removed hash

- [SetChargingProfile.req](#)
 - New
- [SetChargingProfile.conf](#)
 - New
- [StatusNotification.req](#)
 - Overhaul of states
 - New error codes
 - Connector id 0 can only have status: Available, Unavailable and Faulted.
- [StopTransaction.req](#)
 - added explicit and required stop reason
- [TriggerMessage.req](#)
 - New
- [TriggerMessage.conf](#)
 - New
- [UnlockConnector.conf](#)
 - overhaul of [UnlockStatus](#) enum
- [UnitOfMeasure](#)
 - Added Fahrenheit, K, Percent, VA, kVA
 - Rename Volt to V, Amp to A