

OCPP 2.0.1

第 1 部分 - 架构和拓扑

FINAL， 2020-03-31

Git Address: <https://gitee.com/leven9/OCPP-Documentation>

目录

目录

[免責聲明 1](#_Toc139995028)

[版本 历史 2](#_Toc139995029)

[1. 介绍 3](#_Toc139995030)

[1.1. 本文档的目标 3](#_Toc139995031)

[1.2. 术语和缩写 3](#_Toc139995032)

[1.2.1. 条款 3](#_Toc139995033)

[1.2.2. 缩写 3](#_Toc139995034)

[2. 3 层 模型 4](#_Toc139995035)

[3. 信息模型 5](#_Toc139995036)

[4. 设备 型号： 寻址 组件 和 变量 6](#_Toc139995037)

[4.1. 组件 6](#_Toc139995038)

[4.2. 变量 7](#_Toc139995039)

[4.3. 特征 和 属性 7](#_Toc139995040)

[4.4. 监测 10](#_Toc139995041)

[4.5. 组件和变量的标准化列表 11](#_Toc139995042)

[4.6. 最小 设备 型号 11](#_Toc139995043)

[5. 信息 模型 与 设备 模型 12](#_Toc139995044)

[6. 将OCPP用于电动汽车充电以外的其他目的 13](#_Toc139995045)

[7. 编号 14](#_Toc139995046)

[7.1. EVSE 编号 14](#_Toc139995047)

[7.2. 连接器 编号 14](#_Toc139995048)

[7.3. 事务 标识 14](#_Toc139995049)

[8. OCPP 支持的拓扑 15](#_Toc139995050)

[8.1. 直接连接到 CSMS 的充电站 15](#_Toc139995051)

[8.2. 多个充电站 通过本地代理连接到 CSMS 15](#_Toc139995052)

[8.3. 多个充电站 通过本地控制器连接到 CSMS 16](#_Toc139995053)

[8.4. 非 OCPP 充电站通过 OCPP 本地控制器连接到 CSMS 16](#_Toc139995054)

[8.5. DSO 控制 信号 到 CSMS 16](#_Toc139995055)

[8.6. 通过 CSMS 和 EMS 并行控制 17](#_Toc139995056)

[9. 第 1 部分 附录：OCPP 信息模型 18](#_Toc139995057)

[9.1. UML 表示和消息生成的说明 18](#_Toc139995058)

[9.2. OCPP信息模型的可视化表示 19](#_Toc139995059)

# 免責聲明

Copyright c 2010 – 2020 Open Charge Alliance. All rights reserved.

This document is made available under the *\*Creative Commons Attribution-NoDerivatives 4.0 International Public License\**

(https://creativecommons.org/licenses/by-nd/4.0/legalcode).

# 版本历史

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 日期 | 作者 | 描述 |
| 2.0.1 | 2020-03-31 | Franc Buve (*OCA*) Milan Jansen (*OCA*) Paul Klapwijk (*OCA*) | Final version of OCPP 2.0.1 |
| 2.0 | 2018-04-11 | Milan Jansen (*OCA*) Paul Klapwijk (*OCA*)  Robert de Leeuw (*IHomer*) Robben Riksen (*Alliander*) | OCPP 2.0 April 2018  First release of this Architecture & Topology document |

# 介绍

## 本文档的目标

本文档的目标是描述 OCPP 2.0.1 的许多与体系结构相关的主题。

OCPP 最初用于后台，OCPP中的 *充电站管理系统*（在本文档中：CSMS）和充电站之间的双向通信。该协议变得更加先进，并且随着每个新版本的更新，新功能和选项都会添加。它已经发展成为一种协议，可以在不同类型的不同架构中使用充电站。

除了原始的"简单"设置 CSMS <>充电站之外，本文档还介绍了许多拓扑结构，作为使用 OCPP 的附加说明。 此外，设备管理概念可配置和监视任何类型的介绍了充电 站、 OCPP 信息 模型 和 3 层 模型 。

本文件部分**提供信息**，部分**规范，**无意限制OCPP的使用。 但是，它确实增加了一个解释，OCPP 的创建者在创建此版本的规范时想到了 OCPP 的用途。 因此， 本文档还旨在支持 OCPP 第 2 部分中协议规范的读者了解如何使用它。

## 术语和缩写

本节包含本文档中使用的术语和缩写。

### 条款

|  |  |
| --- | --- |
| 术语 | 意义 |
| Charging Station | 充电站是电动汽车可以充电的物理系统。 一个充电站有一个或多个 EVSE。 |
| Connector | 本规范中使用的术语"连接器"是指充电站上独立操作和管理的电源插座。换句话说，这对应于单个物理连接器。在某些情况下，EVSE可能具有多种物理插座类型和/或系留电缆/连接器布置（即连接器），以方便不同的车辆类型（例如四轮电动汽车和电动滑板车）。 |
| EVSE | EVSE被认为是充电站的独立操作和管理的一 部分，可以一次向一辆电动汽车提供能量。 |
| Local port Smart Meter | 智能电表上的本地端口是数字电表上的一个端口（例如串行端口），用于访问有关电表读数和使用情况的信息。 |

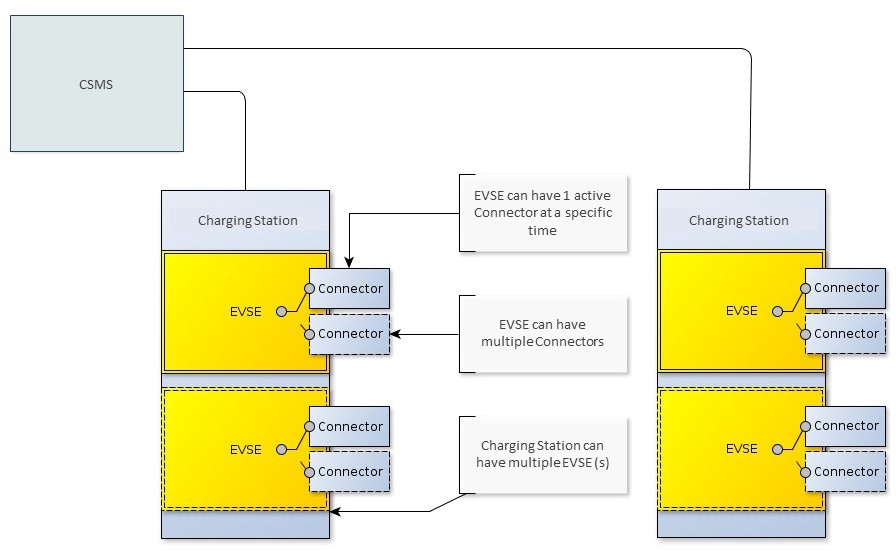
### 缩写

|  |  |
| --- | --- |
| 缩写 | 意义 |
| DSO | 配电系统操作员 |
| CSO | 充电站操作员 |
| CSMS | 充电站管理系统 |
| EMS | 能源管理系统。在本文档中，这被定义为根据本地和/或合同约束和/或合同激励措施管理本地负载（消耗和 生产）的设备。 它具有 额外的输入， 例如来自PV，电池存储等的传感器和控制。 |
| EVSE | 电动汽车供应设备 |
| LC | 本地控制器。在本文档中，这被定义为可以独立于 CSMS 向其充电站发送消息的设备。典型的用法是 OCPP 第 2 部分的智能充电章节中描述的本地智能充电案例，其中本地控制器可以对其充电站施加充电限制 。 |
| LP | 本地代理。 充当消息路由器。 |

# 3 层模型

本节内容丰富。

要理解 OCPP 规范中的术语，请务必了解此规范的起点 。 OCPP规范使用术语充电站作为可以为电动汽车充电的物理系统。 一个充电站可以有一个或多个 EVSE（电动汽车供应设备）。EVSE 被认为是充电站的一部分，可以一次向一辆电动汽车提供能量。本规范中使用的 term 连接器是指充电站 上独立操作和管理的 电源插座，换句话说，这对应于单个物理连接器。在某些情况下，EVSE 可能具有多种物理插座类型和/或系留电缆/连接器布置，以方便不同的车辆类型（例如四轮电动汽车和电动滑板车）。 此设置称为 3 层模型，如下图所示。



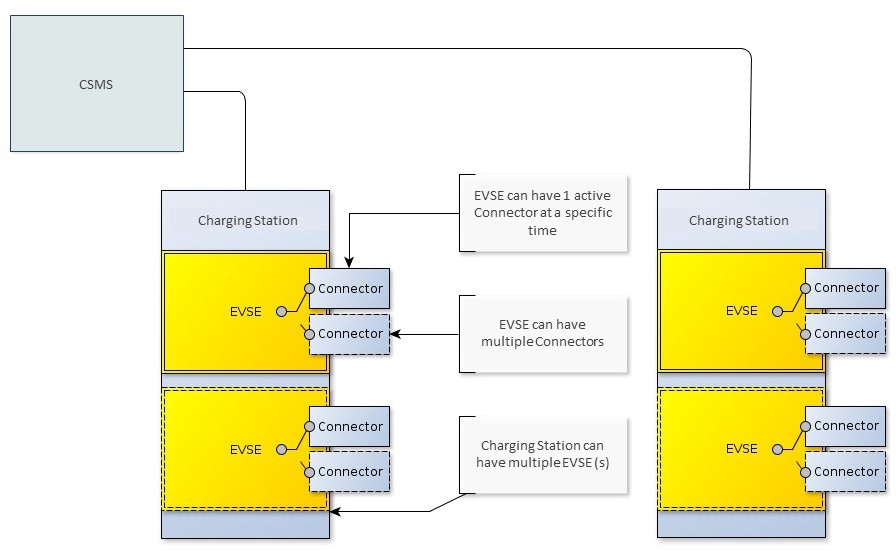


图 1. OCPP 中使用的 3 层模型

注意

本节介绍用于通信目的的逻辑级别的充电基础结构。 我们不希望将映射强加到物理硬件上。这是制造商的选择。例如，EVSE 可以集成到充电站中，并且看起来只是该设备的一部分，但它也可能有自己的外壳，并且位于物理实体充电站之外，例如具有 20 个 EVSE 和连接器的充电广场，它通过 1 个调制解调器作为 1 个充电站与 OCPP 将 CSMS 视为 1 个充电站。

# 信息模型

本节内容丰富。

鉴于 OCPP 消息的复杂性日益增加，OCPP 2.0.1 基于*信息模型*，作为 OCPP 消息和固有模式的蓝图。 对于信息模型，我们指的是一个逻辑对象集，描述具有其所有属性的真实对象。 这提供了协议中信息结构的信息表示。 此外，它使 OCPP 中的对象可重用，并实现消息和自动生成消息的一致定义架构 （第 3 部分）。

信息模型是一个模型，也称为域模型或核心模型，基于该模型生成 OCPP 消息和数据类型。这些数据类型是从 OCPP 1.6 规范中提取的， 命名为核心数据类型和合格数据类型。 下图说明了如何构建信息模型中的数据类型。

在第 2 部分 - 规范 DataTypes 一章中，某些数据类型具有 Common: prefix。 这源于信息模型。 这意味着数据类型能够在其他数据类型和消息之间共享。这对设备的 OCPP 实现没有影响 。

*anySimpleType*

«XSDsimpleType»

**XSDDatatypes::decimal**

«PRIMSimpleType»



*string*

«XSDsimpleType»

**XSDDatatypes::normalizedString**

*(from PrimitiveDataTypes)*

«Supplementary»

+ identificationScheme: String [0..1]

+ identificationScheme.Agency: String [0..1]

+ identificationScheme.AgencyName: String [0..1]

+ identificationScheme.Name: String [0..1]

+ identificationScheme.UniformResource: String [0..1]

+ identificationScheme.Version: String [0..1]

.1] + identificationSchemeData.UniformResource: String [0..1]

*(from CoreDataTypes)*

}

«Facet»

+ maxLength = 20 {readOnly

«BDTComplexType»

**CertificateSerialNumberType**

«CDTComplexType,LDT,ECDMChange,CDT»

**IdentifierType**

«PRIMSimpleType»

**NormalizedString**

**Decimal**

*(from PrimitiveDataTypes)*

«CDTComplexType,ECDMChange,...

«Supplementary»

+ measureUnit: String [0..1]

+ measureUnit.CodeListVersion: String [0.

«CDTComplexType,LDT,ECDMChange,...

**MeasureType**

«CDTSimpleType,LDT,EC...

**NumericType**

**AmountType**

«Supplementary»

+ format: String [0..1]

*(from CoreDataTypes)*

«Supplementary»

+ currency: String [0..1]

+ currencyCodeListVersion: String [0..1]

*(from CoreDataTypes)*

*(from CoreDataTypes)*

«BDTSimpleType»

«BDTSimpleType»

**IntegerType**

**PercentageType**

«BDTComplexType»

**PositiveAmount**

«BDTComplexType»

**PowerType**

}

«Facet»

+ fractionDigits= 0 {readOnly

*(from QualifiedDataTypes)*

«Facet»

+ fractionDigits= 0 {readOnly}

+ minInclusive = 0 {readOnly}

+ maxInclusive = 100 {readOnly}

*(from QualifiedDataTypes)*

«Facet»

+ fractionDigits= 0 {readOnly}

+ minInclusive = 0 {readOnly}

*(from QualifiedDataTypes)*

*(from QualifiedDataTypes)*

*(from QualifiedDataType*

图 2. 示例数据类型

信息模型分为许多 "功能"，以便更好地概述模型（因此是为了便于阅读）：

* 交易
* 智能充电
* 计量
* 安全性 （配置文件/授权）
* 通信
* SecondaryActor Scheduleule

有关每个函数的实际型号的更多详细信息，请参阅附录。

# 设备型号：寻址组件和变量

设备模型是指 OCPP 中的一种通用机制，使任何型号的充电站都能报告其如何构建，因此可以从任何 CSMS 对其进行管理。为了使用设备模型管理充电站（即"管理设备"），定义了许多消息和用例来详细配置和监控充电站，而无需事先定义充电站的结构 。 为了能够做到这一点，OCPP 提供了一种通用机制，允许交换有关充电站的各种信息。此版本的设备型号以 3 层模型（充电站、EVSE、连接器）为起点，这意味着使用设备型号创建的任何描述都遵循这三层。本章的重构者描述了数据（和相关的元数据）在充电站和 CSMS 之间交换的样子。此处未介绍用于管理设备的用例和消息 ，但在本规范的第 2 部分中进行了介绍。 本章仅重点介绍数据模型。

## 组件

在 OCPP 2.0.1 中，充电站被建模为一组*"组件"*，通常表示物理设备（包括它连接到的任何外部设备以进行数据收集和/或控制）、逻辑功能或逻辑数据实体。

不同类型的组件主要由组件名称 标识，该组件名称是*标准化* 组件的名称 （请参阅 OCPP 第 2c 部分），或自定义/非标准化组件名称，用于 新的预标准化设备，供应商特定扩展等。

充电站（TopLevel）、*EVSE*和  *Connection*代表充电站的三大 "层"，构成隐式 "基于位置" 的寻址 方案 ， 广泛 用于许多 OCPP 数据结构。

默认情况下，所有*组件*都位于*充电站*层，但任何组件的单个实例都可以与特定的  *EVSE* 或 特定的*连接器*（在特定的 EVSE 上）相关联。 通过将 EVSE 或 EVSE 和连接器标识号作为组件寻址引用的一部分。

此外， 一个组件可以有多个实例（在功能维度中），表示多重出现 物理 或 逻辑 组件 （例如 电源 转换器 模块、 风扇 组、 驻留 固件 映像 等）。

每个不同的*组件*实例都 由一个（可选）*组件实例*寻址键唯一标识。当否*提供*组件实例，然后引用 *组件*的默认实例或仅实例。

*组件* 本身不 保存数据：与每个组件实例 关联的所有外部可访问数据都由一组变量表示，这些*变量*可以读取、设置和/或监视 更改。 组件与 一 个或多个变量的关系如下图所示。

0

1

1

\*

+ Instance: NameIdentifierType [0..1]

+ Name: NameIdentifierType

«BusinessComponent»

**VariableType**

+ Instance: NameIdentifierType [0..1]

+ Name: NameIdentifierType

+ ConnectorId: NumericIdentifierType [0..1]

«Content»

+ MRID: NumericIdentifierType

«BusinessComponent»

**EVSEType**

«BusinessComponent»

**ComponentType**

图 3. 组件和变量

下表说明了一些常见组件（通过其标准化组件名称），以及基本家用充电器通常出现的分层位置级别的示例。和一个典型的公共充电站。

|  |  |  |
| --- | --- | --- |
| 基本家用充电器示例配置 | | |
| 充电站层 | 云服务器层 | 连接器层 |
| 充电站 （本身， 作为 一个 整体） | EVSE （本身， 作为 一个 整体） | 连接器 （本身， 作为 一个 整体） |
| 无线电链接 | 控制计量 | PlugRetentionLock |
| 令牌读取器 | 过电流断路器 |  |
| 控制器 | 刚果民盟 |  |
|  | 充电状态指示器 |  |

|  |  |  |
| --- | --- | --- |
| 公共 充电站 示例配置 | | |
| 充电站 层 | 云服务器层 | 连接器 层 |
| 充电站 （本身， 作为 一个 整体） | EVSE （本身， 作为 一个 整体） | 连接器 （本身， 作为 一个 整体） |
| 电气馈电 | 电气馈电 | 访问保护 |
| 令牌读取器 | 令牌读取器 | PlugRetentionLock |
| 显示 | 显示 |  |
| 财政计量 | 财政计量 |  |
| 时钟 | 控制计量 |  |
| 控制器 | 过电流断路器 |  |
|  | 刚果民盟 |  |
|  | 充电状态指示器 |  |

## 变量

每个*组件*都有许多*变量*，这些变量可以根据需要用于保存、设置、读取和/或报告适用于 该*组件*的所有（外部可见）数据，包括配置参数、测量值（例如 电流或温度）和/或监视变量值的更改。

尽管许多*组件*可以具有特定于组件类型的关联*变量*（例如，*连接器*组件的  *ConnectorType*），但有一组最小的标准化*变量*可用于提供标准化的高级事件通知和状态/状态报告（例如*，问题*、*活动*）） 全局和/或有选择地报告组件的存在、可用性等（例如，*可用*、*已启用*）。当 基本变量被读并设置为  *true* 时，不需要充电暂停来报告 基本变量："*存在*"、"*可用*  *"和"已启用"*。 充电时 工作站未 报告：*存在*、*可用*和/或*已启用* 中央系统应假定它们仅被读取并设置为  *true* 变量可以是 一系列 常见通用 数据类型（布尔值、整数、十进制、日期时间、字符串）中的任何一个，但也可以将其允许的值限制为特定范围、枚举列表、集或有序列表。

为了支持复杂的组件， 可以有多个实例，任何给定的变量名称与任何组件（例如，电源转换器模块报告多个点的温度、电流或电压）。

每个不同的*变量*实例都 由一个（可选）*变量实例*寻址键字符串值唯一标识。 当否提供变量实例，然后引用  *变量*的默认实例或仅实例。

## 特征和属性

每个*变量* 除了 其主变量（*"实际"*）值外，还可以具有一组 关联的辅助数据，这些数据 链接到 相同的主*变量*名称和*变量Instance*.

这大大避免了 变量命名空间与令人困惑的辅助变量名称簇（例如FanSpeed，FanSpeedUnits， MinimumFanSpeed， BaseFanSpeed） 缺乏 一致性 和 可发现性。 辅助 变量 数据 包括：

* 可变 特征 元数据 （只读）
  + 计量 单位 （V、W、kW、kWh 等）
  + 数据类型（整数、十进制、字符串、日期、选项列表等）
  + 下 限
  + 上限
  + 枚举变量的允许值列表
* 变量属性 （读写）：
  + 实际 值
  + 目标 值
  + 配置 的下限
  + 配置的 上限
  + 可变性（无论 是否可以 更改值 ，例如"只读"或"读写"）
  + 持久性（在 重新启动或断电时 是否保留该值 ）

下图 说明了 变量与一 个或多个变量属性的关系。

0

1

1

\*

]

**1**

1

1..\*

0..1

+ Constant: IndicatorType

+ Mutability: MutabilityEnumType [0..1]

+ Persistent: IndicatorType

+ Type: AttributeEnumType [0..1]

+ Value: Base64String2500Type

«BusinessComponent»

**VariableAttributeType**

+ DataType: DataEnumType

+ MaxLimit: AmountType [0..1]

+ MinLimit: AmountType [0..1]

+ SupportsMonitoring: IndicatorType

+ Unit: MeasurementUnitType [0..1]

+ ValuesList: CI1000TextType [0..1]

«BusinessComponent»

**VariableCharacteristicsType**

+ Instance: NameIdentifierType [0..1

+ Name: NameIdentifierType

«BusinessComponent»

**VariableType**

+ ConnectorId: NumericIdentifierType [0..1]

«Content»

+ MRID: NumericIdentifierType

+ Instance: NameIdentifierType [0..1]

+ Name: NameIdentifierType

«BusinessComponent»

**EVSEType**

«BusinessComponent»

**ComponentType**

图 4. 可变 属性和特征

使用 DeviceModel 实现（物理） 设备和（虚拟）控制器组件之间存在差异。 必须实现（虚拟）控制器组件 ，如 第 2 部分"引用的组件和变量"一章所述。 这些类型的组件/变量仅使用变量属性类型"实际"。根据此变量属性是否可写， CSMS 可以使用 它来设置 新值。

（物理） 设备实现起来有点 复杂。例如，有一个风扇速度的风扇， 其（物理）限制范围为 0 - 1000。 但 不应 允许将值设置为 低于200，因为风扇可能会停止运行。而且它不应该 设置在 500 以上，因为从长远来看，这对风扇来说是不利的。 使用 DeviceModel 实现此设备时，可以 按如下方式定义它：

|  |  |  |  |
| --- | --- | --- | --- |
| 元件 | 名字 | 扇 | |
| 变量 | 名字 | 风扇速度 | |
| 变量属性 1 | 类型 | 当前 |
| 价值 | <风扇的当前风扇转速值> |
| 易变性 | 只读 |
| 变量属性 2 | 类型 | 目标 |
| 价值 | < CSMS 可以使用此值来调整风扇速度。 充电 站应尽量将实际值保持在目标值> |
| 易变性 | 读写 |
| 变量归因 3 | 类型 | 美司特 |
| 价值 | <示例 中的值"500"。 目标 不能设置在此值之上。> |
| 变量属性 4 | 类型 | 最小集 |
| 价值 | <示例 中的值"200"。 目标 不能 设置在此值以下 。> |
| 变量特征 | 最大限度 | < 示例中的值"1000" 。 这可能是风扇的物理最大限制 >。 |
| 最小限度 | <示例中的值"0"。这可能是风扇的物理最小 值限制。 如果 风扇 也能够向 另一个方向旋转，则此值 也可能是 -1000。> |
| 描述 | 这是如何使用 DeviceModel 定义 风扇 的示例。 | | |

尝试将目标设置为值 600 时，充电站将首先检查允许的最小值和最大值/限制并拒绝设置。 如果目标值设置为 500，则该值 在范围内，充电站将允许设置 并开始调整实际的风扇速度。 如果实际风扇速度测量为 502，则超出范围。 但它应该报告给 CSMS，所以 应更新物理组件的实际 值，而不检查最小值和最大值/限制。

## 监测

可选的监视设置 可以 与 变量相关联，该变量允许将*变量*（*实际*）值的更改作为事件通知报告给 CSMS。

这些 包括：

* 监控 价值
* 监控 类型： 阈值上限 、 阈值下限 、 增量、 周期性
* 报告事件时的严重性级别

下表显示了哪些 MonitorType/dataType 组合 是可能的。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 字符串 | 十进制 | 整数 | 日期时间 | 布尔 | 选项列表 | 序列列表 | 会员名单 |
| 上雷什 旧 |  | X | X |  |  |  |  |  |
| 下雷什 旧 |  | X | X |  |  |  |  |  |
| 三角洲 | X | X | X | X | X | X | X | X |
| 周期的 | X | X | X |  | X | X | X | X |
| 周期性位置 k对齐 | X | X | X |  | X | X | X | X |

* 对于*上阈值*和*下阈值* ，该值表示 变量 的实际值将超出的值 。
* 对于  *Delta*，此值表示与设置监视器时的实际值相比 ，值的变化。
  + 当 变量 的数据类型为整数或十进制时，此值表示触发监视器时要 达到的差异。
  + 当 变量的 数据类型为 dateTime 时， 度量单位将 以秒为单位。
  + 当 变量的数据类型 为字符串、布尔值、选项列表、序列列表或成员列表时，将忽略此值。 监视器 将由 实际值的每次更改触发。
* 当触发增量监视器 或充电站重新启动时，充电站应设置 新的瞬时值。
* 对于*"定期*"和*"定期时钟对齐"*，该值表示以秒为单位的间隔。

下 图 说明了 变量与一 个或多个变量监视元素之间的关系。

0

1

1

\*

1

1

1

1..\*

0..1

\*

+ Constant: IndicatorType

+ Mutability: MutabilityEnumType [0..1]

+ Persistent: IndicatorType

+ Type: AttributeEnumType [0..1]

+ Value: Base64String2500Type

«BusinessComponent»

**VariableAttributeType**

+ Id: NumericIdentifierType

+ Severity: NumericIdentifierType

+ Transaction: IndicatorType

+ Type: MonitorEnumType

+ Value: MeasureType

+ DataType: DataEnumType

+ MaxLimit: AmountType [0..1]

+ MinLimit: AmountType [0..1]

+ SupportsMonitoring: IndicatorType

+ Unit: MeasurementUnitType [0..1]

+ ValuesList: CI1000TextType [0..1]

«BusinessComponent»

**VariableMonitoringType**

«BusinessComponent»

**VariableCharacteristicsType**

+ Instance: NameIdentifierType [0..1]

+ Name: NameIdentifierType

«BusinessComponent»

**VariableType**

+ ConnectorId: NumericIdentifierType [0..1]

«Content»

+ MRID: NumericIdentifierType

+ Instance: NameIdentifierType [0..1]

+ Name: NameIdentifierType

«BusinessComponent»

**EVSEType**

«BusinessComponent»

**ComponentType**

*Figure 5. Variables and monitoring*

## 组件和变量的标准化列表

为了在不同的充电站和CSMS之间提供一定程度的互操作性，除了上面定义的*组件*和*变量*模型之外，OCPP规范的第2部分 - 附录提供了组件和可变参数的标准化名称列表。此列表的想法是确保*如果*充电站和 CSMS 想要交换有关组件的信息，*则*在 OCPP 规范中列出时，它们都使用相同的名称和描述。对于 规范中 未列出的*组件*或*变量*的名称，充电站制造商和CSMS之间的双边约会应为 䍬。在这些情况下，建议向开放收费联盟提供反馈，以便能够在 新版本的 OCPP 中包含新的/额外的*组件*和*变量*。

## 最小设备型号

由于设备模型是一种*通用*机制，可以应用于任何型号的充电站，因此不同实现的复杂性可能会有所不同。它由许多用例和消息组成，这些用例和消息并非全部必需。本节介绍 创建 OCPP 2.0.1 的工作实现 时需要实现 的设备模型的最小部分。

设备模型引入了可用于配置和监视充电站的组件和变量。 这些组件和变量中的 许多都包含在规范第 2 部分 的*"参考组件和变量"*列表中（按功能块分组）。实现功能块时， 应实现属于功能块的所有必需配置变量。  *"常规*"部分中所需的配置变量 也应针对 OCPP 2.0.1 的所有实现实现。

下表描述了对于作为设备模型 实现一部分的所有用例 ，哪些消息是必需的或可选的。

|  |  |
| --- | --- |
| 作为最小设备模型实现一部分的用例/消息 | |
| 使用案例 | 消息 |
| B05 设置 变量 | 必须 实现变量消息 |
| B06 获取 变量 | 必须 实现 GetVariables 消息。 |
| B07 获取 基本 报告 | GetBaseReport 消息必须 实现，并且必须支持所有 3 个"报告库"。 这些报告 的内容取决于 充电站 的实施情况。 由 实现者决定 实现中存在 哪些组件和变量。 |
| *不属于* 最小设备模型实现 的其他 用例/消息 | |
| 使用案例 | 消息 |
| B08 获取 自定义 报告 | GetCustomReport 消息 是 可选的。 |
| N02 获取监控报告 | GetMonitoringReportRequest 消息 是 可选的。 |
| N03 集 监控 库 | SetMonitoringBaseRequest 消息 是 可选的。 |
| N04 设置 变量 监控 | 设置可变监视请求 消息是可选的。 |
| N05 设置 监控 电平 | SetMonitoringLevelRequest 消息 是 可选的。 |
| N06 清除/删除 监控 | 清除可变监视请求 消息 是 可选的。 |
| N07 警报 事件 | 建议即使在未实施监视时，也应在充电站中实现 NotifyEventRequest，以便 可用于报告内置监视事件。 |
| N08 定期 事件 | 见 N07。 |

# 信息模型与设备模型

如上所述，术语"信息模型"和"设备模型" 指的是不同的概念。信息模型是指 OCPP 中的消息和数据类型所基于的信息结构模型，而设备模型是指 OCPP 中的广义机制，使充电站的任何模型都能报告它是如何建立起来的，所以，它可以被管理从任何CSMS，无需事先定义充电站的结构。

因此，用于设备管理的消息是 信息模型和 用于设备建模的对象（*"组件"*和"*变量"*）的一部分。 也是信息模型的一部分。

# 将OCPP用于电动汽车充电以外的其他目的

如本文档简介所示，OCPP 主要用于 CSMS 和充电站之间的双向通信 。 但是， 随着设备 [Model](#_bookmark7) 一章中所述[的设备](#_bookmark7)模型的添加，OCPP 还可以用于其他目的。例如，报告变压器或独立电池组中的事件或状态变化 也可能对 正在推出 电动汽车充电基础设施的公司有用。 在此示例中，引导通知 可用于 将这些设备连接到管理系统。在设备型号中，可以通过组件充电站不在顶层来识别不是充电站的设备。目前，OCPP规范没有为非充电站设备提供用例。 但是，它们 可能会添加到 OCPP 的未来版本中。

# 编号

本节是规范性的。

## EVSE 编号

要使 CSMS 能够处理 充电站的所有 EVSE，EVSE 必须始终 以相同的 方式编号。 EVSE 编号 （evseIds） 必须 如下所示：

* EVSE 必须按顺序编号，从每个充电站的 1 开始 （不得跳过任何数字）。
* evseIds 不得 高于 充电站的 EVSE 总数
* 对于由 CSMS 启动的操作，evseId 0 保留用于寻址整个 充电站。
* 对于由充电站启动的操作（报告时），evseId 0 保留给 充电站主控制器。

示例： 具有 3 个 EVSE 的充电站：所有 EVSE 必须 使用 ID 编号：1、2 和 3。 建议以 合乎逻辑的方式对 充电站 的 EVSE 进行编号：从左到右，从上到下递增。

## 连接器编号

要使 CSMS 能够对 充电站 的所有连接器进行寻址，连接器必须始终以相同的方式编号。 连接器编号（连接器 Id）必须 如下所示：

* 连接器 从每个 EVSE 上的连接器 Id 1 开始编号（递增）
* 每个 EVSE 的每个连接器都有一个唯一的编号
* EVSE 的第一个连接器的 ID 必须为 1
* 同 一 EVSE 的附加连接器 必须按顺序编号（不得跳过任何编号 ）
* connectorIds 不得高于该 EVSE 上的连接器总数 示例：具有 3 个 EVSE 且每个 EVSE 有 2 个连接器的充电站的编号如下：
* EVSE 1 具有带连接器 Id 1 和 2 的连接器
* EVSE 2 具有 带连接器 Id 1 和 2 的连接器
* EVSE 3 具有 带连接器 Id 1 和 2 的连接器

## 事务标识

TransactionId 现在由充电站生成，并且对于每个已启动的事务，该充电站上必须是唯一的。

在 OCPP 1.x 中，这是由CSMS完成的。

事务 ID 的格式 留待 实现。 例如 ，这可能是 一个增量数字或一个UUID。

# OCPP 支持的拓扑

本章介绍 使用 OCPP 的多种拓扑。 如简介中 所述 ，OCPP 最初 用于 每个充电站直接与CSMS通信的设置。 重要的是要 记住，OCPP 不 了解充电站网络的 拓扑结构。 下图显示了 使用 OCPP 的设置中可能的组件 以及这些组件之间的关系：

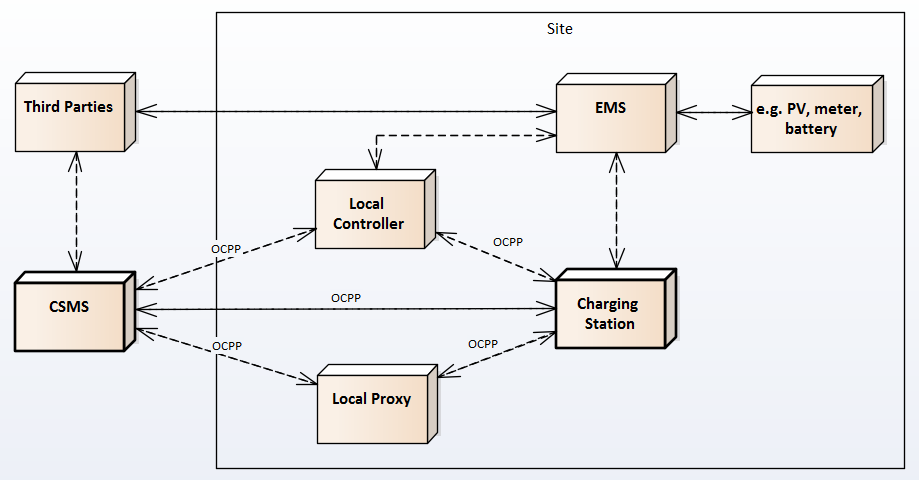


图 6. 使用 OCPP 的设置中可能的组件

## 直接连接到 CSMS 的充电站

描述

这是 使用 OCPP 的基本设置。

断续器

断续器

充电 站

图 7. 直接 连接到 CSMS 的充电站

## 多个充电站通过本地代理连接到 CSMS

描述

在某些情况下 ，需要通过 单个网络节点（即 调制解调器，路由器等）路由一组充电站的所有通信。一个典型的例子是，许多充电站位于地下停车场 ，很少或根本无法访问 移动网络。 为了提供 对移动数据的访问，充电站通过LAN连接到中央数据通信单元。该中央单元连接到移动网络，并充当CSMS和充电站之间的代理。 这样的 单元在 OCPP 中称为 "本地代理"（LP）。 本地代理 充当 消息路由器。 CSMS和 充电站都不知道 网络的拓扑 结构。 对于 组中的充电站 ，本地代理"是"CSMS。同样，对于CSMS，本地代理"是"充电站。下图说明了此配置。

断续器

充电 站1

断续器

断续器

断续器

断续器

充电 站 n

图 8. 多个充电站通过本地代理连接到 CSMS

## 多个充电站通过本地控制器连接到 CSMS

描述

[本地代理](#_bookmark22) 仅执行 路由 OCPP 消息，而本地控制器可以独立于 CSMS 向其充电站发送消息。典型的用法是 OCPP 第 2 部分的智能充电章节中描述的本地智能充电案例，其中本地控制器可以对其充电站施加充电限制。为了使本地控制器由 CSMS 寻址，它需要有自己的充电站标识。从 OCPP的角度来看，本地控制器 将只是 一个充电站（没有任何 EVSE/连接器）。 CSMS将拥有 处理本地控制器 的逻辑，以支持例如本地智能充电。这取决于 CSMS 的实现，无论组拓扑是手动配置的，还是 根据 BootNotifications 中的 IP 地址和信息从网络推断出来的。 下图 说明了此配置。

断续器

充电 站1

断续器

断续器

立法会

断续器

充电 站n

图 9. 多个充电站 通过本地控制器连接到 CSMS

注意

从技术上讲，这种拓扑可以通过多种方式实现。将此设置与 websocket 一起使用时，这意味着当 充电站 连接到 本地控制器时， 它应该打开与 CSMS 具有相同地址的 websocket 连接。这种方法的缺点是，本地控制器可以看到所有消息并对其执行操作，消息不必等待，充电站上的固件更新等是可能的，并且CSMS不需要特殊的软件。它（在大型安装中）可能导致CSMS和LC之间需要很多websocket连接。 有关详细信息， 请参阅第 4 部分中的 OCPP 实施指南。

## 非 OCPP 充电站通过 OCPP 本地控制器连接到 CSMS

描述

此设置具有多个非 OCPP 充电站，这些充电站 使用启用了 OCPP 的本地控制器进行抽象。 在这种情况下应用 OCPP 时 ，应将 LC 视为 具有许多 EVSE 的充电站，或者 LC 应充当多个 OCPP 充电站 （ 拥有自己的 充电站标识）。

非超频

充电 站1

断续器

断续器

立法会

非超频

充电 站n

图 10. 多个非 OCPP 充电站 通过本地控制器连接到 CSMS

## DSO 控制信号到 CSMS

描述

在这种设置中，CSMS是唯一向其充电站发送信号的应用程序，但CSMS根据（最容易的）电网约束从DSO接收智能充电信号。这意味着接收到非OCPP信号，如OpenADR或OSCP ，并且基于 此信号， CSMS限制其充电站的充电。 希望完全控制其充电站 的 CSO使用此 体系结构，这样他们就可以控制 其充电 所使用的能量。站。 这 可以通过向充电站发送充电配置文件/充电时间表 来完成。

非超频 （例如 打开广告 或 OSCP） 断续器

断续器

断续器

充电 站

图 11. 智能 充电 - DSO 控制 信号 到 CSMS

## 通过 CSMS 和 EMS 并行控制

描述

在（半）私人情况下，充电站不仅连接到CSMS，而且还连接到能源管理系统， 应该 支持某种形式的 并行控制。 OCPP 至少应该用于 充电站维护，但 OCPP 2.0.1还支持报告外部智能充电控制限制。因此，如果能源管理系统决定以后充电"更好"，则能源管理系统可以对充电站施加外部限制（例如0）， 充电站 又可以通过 OCPP向 CSMS报告。 能源管理系统 可能会从 智能电表的本地端口获得输入，以防止连接过载，但也可能有其他原因不充电（例如天气条件）。

断续器 断续器 或 其他

断续器

特快专递

充电 站

图 12. 通过 CSMS 和 EMS 并行控制

# 第 1 部分 附录：OCPP 信息模型

## UML 表示和消息生成的说明

在下一段中，显示了 OCPP 信息模型的 UML 方案。该模型基于通用信息模型（CIM），并在 一定程度上基于 CEFACT命名标准（仅是标准的一部分）。 模型 中的对象名为*"业务组件"，*并从 CIM *标识*对象继承属性，如"MRID"和"名称"。在 UML 关系图中， 从*已识别对象* 继承的属性显示在"已*识别*对象"构造型下（在<<>>之间）。

其他属性列在内容>>< 构造型<下。

OCPP 中 的消息派生自 下一段中表示的模型，分为 3 个步骤：

信息 模型



消息 模型

规范

数字 13. 过程 从 信息 型 自 消息 / 计划



模式

创建 信息模型后 ， 将基于 信息模型创建消息。 但是， 在此转换（第一个箭头）中，某些规则（手动）应用于建模消息。应用的最重要规则是，包含对 仅 具有一个 <字段><类>的引用的消息将替换为名称为<类><字段>的字段。 例如，如果

message 包含一个事务，只有一个 Id，这 被一个事务 Id 所取代。

在下一步中，当生成规范第 2 部分的消息和数据类型部分时，为了便于阅读，所有核心 数据类型 （如 *CounterType*） 都将替换为它们在此示例 *integer 中*引用的基元数据类型（枚举除外）。

## OCPP信息模型的可视化表示



«BusinessComponent»

**Components::IdentifiedObject**

+TxnStartSampledValue

«BusinessComponent»

**Metering::SampledValueType**

0..\*

«BusinessComponent»

**Components::MessageInfoType**

«Content»

+ MRID: NumericIdentifierType [0..1]

+ ObjectID: IdentifierType [0..\*]

+ Name: NameType [0..1]

+ AliasName: NameType [0..\*]

+ Description: TextType [0..\*]

«Content»

+TxnStopSampledValue+ Context: ReadingContextEnumType [0..1]

+ Value: MeasureType [0..1]

+ Phase: PhaseEnumType [0..1]

+ Location: LocationEnumType [0..1]

+TxnUpdateSampledValue

0..\* + Measurand: MeasurandEnumType [0..1]

«Content»

+ Priority: MessagePriorityEnumType [0..1]

+ State: MessageStateEnumType [0..1]

+ StartDateTime: DateTimeType [0..1]

+ EndDateTime: DateTimeType [0..1]

+Message

0..\*

0..\*

+Transaction

«BusinessComponent»

**TransactionType**

*DeviceType*

0..1

*DeviceFunctionType*

«BusinessComponent»

**Metering::MeteringFunctionType**

«Content»

+ SeqNo: CounterType [0..1]

«BusinessComponent»

**RootModel::EVSEType**

+ ConnectorId: NumericIdentifierType [0..1]

+EVSE «Content»

+EVSE

+MeteringFunction

0..\*

+Transaction + ChargingState: ChargingStateEnumType [0..1]

+ Offline: IndicatorType [0..1]

0..\* + StartedTimestamp: DateTimeType [0..1]

+ StoppedTimestamp: DateTimeType [0..1]

+ TimeSpentCharging: SecondsType [0..1]

+Transaction

0..1

0..\*

0..1 + Switch3To1PhaseSupported: IndicatorType [0..1]

+EVSE 0..1

«BusinessComponent»

**RootModel::EVType**

+EV

+Transaction + StoppedReason: ReasonEnumType [0..1]

+ NumberOfPhasesUsed: CounterType [0..1]

0..1 0..1 + MaxAllowedHours: SecondsType [0..1]

+ MaxAllowedKwh: PowerType [0..1]

+EV 0..\*

+Transaction + TriggerReason: TriggerReasonEnumType [0..1]

+Transaction

0..\*

0..1 + RemoteStartID: RequestIdType [0..1]

+ Id: TransactionIdType [0..1]

+ RunningCost: AmountType [0..1]

+ TotalCost: AmountType [0..1]

+Reservation 0..\*

+Transaction 0..1

+Transaction 0..1

+Reservation

+ChargingProfile 0..\*

«BusinessComponent»

**ReservationType**

0..1

«BusinessComponent»

**SmartCharging::ChargingProfileType**

+IDToken 0..\*

«Content»

+ CreationTimestamp: DateTimeType [0..1]

+Reservation + ExpiryDateTime: DateTimeType [0..1]

+ TransactionId: TransactionIdType [0..1]

«Content»

+ StackLevel: CounterType [0..1]

+ Primary: IndicatorType [0..1]

+ TimeBase: DateTimeType [0..1]

+ ChargingProfilePurpose: ChargingProfilePurposeEnumType [0..1]

+ ChargingProfileKind: ChargingProfileKindEnumType [0..1]

+ RecurrencyKind: RecurrencyKindEnumType [0..1]

+ ValidFrom: DateTimeType [0..1]

+ ValidTo: DateTimeType [0..1]

+ChargingProfile

«BusinessComponent»

**Security::IDTokenType**

+IDToken

0..1

*::IdentifiedObject*

0..1

+ IDToken: TokenType [0..1]

+ Type: IDTokenEnumType [0..1]

0..\* + MRID: NumericIdentifierType [0..1]

+ ObjectID: IdentifierType [0..\*]

+GroupIDToken +Reservation + Name: NameType [0..1]

0..1

0..\* + AliasName: NameType [0..\*]

+ Description: TextType [0..\*]

+IDToken 0..1

+Reservation 0..\*

+IDTokenInfo 0..1

+ ChargingLimitSource: ChargingLimitSourceEnumType [0..\*]

«BusinessComponent»

**Security::IDTokenInfoType**

+Connector 0..1

*DeviceType*

+ChargingProfile 0..\*

«BusinessComponent»

**RootModel::ConnectorType**

+Connector

«Content»

+ Status: AuthorizationStatusEnumType [0..1]

+ CacheExpiryDateTime: DateTimeType [0..1]

+ Language1: LanguageType [0..1]

+ Language2: LanguageType [0..1]

+ ChargingPriority: NumericIdentifierType [0..1]

+ ListID: IdentifierType [0..1]

+ ListEntryTimestamp: DateTimeType [0..1]

0..1

«Content»

+ ConnectorType: ConnectorEnumType [0..1]

+ CableCapacity: CurrentType [0..\*]

*Figure 14. OCPP Information Model: Transactions*

«Content»

«BusinessComponent»

**ChargingLimitType**

«BusinessComponent»

**ChargingSchedulePeriodType**



«Content»

+ StartPeriod: SecondsType [0..1]

+ChargingLimit+ Limit: SingleFractionDigitNumericType [0..1]

+ ChargingRateUnit: ChargingRateUnitEnumType [0..1]

0..1 + ChargingLimitSource: ChargingLimitSourceEnumType [0..1]

+ IsGridCritical: IndicatorType [0..1]

+ Limit: MeasureType [0..1]

+ NumberPhases: CounterType [0..1]

+ PhaseToUse: NumericType [0..1]

+ChargingSchedulePeriod 0..\*

«BusinessComponent»

**CompositeScheduleType**

+ChargingSchedu 0..\*

*IdentifiedObject*

le

+CompositeSchedule «Content»

0..\* + Duration: SecondsType [0..1]

+ ChargingRateUnit: ChargingRateUnitEnumType [0..1]

+ChargingSchedule

0..1

«Content»

«BusinessComponent»

**ChargingScheduleType**

+EVSE 0..\*

+EVSE 0..1

*DeviceType*

+ StartDateTime: DateTimeType [0..1]

+CompositeSchedule 0..\*

+ StartSchedule: DateTimeType [0..1]

+ Duration: SecondsType [0..1]

+ ChargingRateUnit: ChargingRateUnitEnumType [0..1]

«BusinessComponent»

**RootModel::EVSEType**

+ ConnectorId: NumericIdentifierType [0..1]

«BusinessComponent»

*IdentifiedObject*

+ MinChargingRate: NumericType [0..1]

+ChargingSchedule 0..1

«Content»

+ Switch3To1PhaseSupported: IndicatorType [0..1]

«Content»

**Transactions::TransactionType**

+ChargePoint0..1

*DeviceType*

++EVSE 00..1

EVSE

..1

+EVSE 0..1

+ SeqNo: CounterType [0..1]

+ ChargingState: ChargingStateEnumType [0..1]

+ Offline: IndicatorType [0..1]

+ StartedTimestamp: DateTimeType [0..1]

+ StoppedTimestamp: DateTimeType [0..1]

+ TimeSpentCharging: SecondsType [0..1]

+ StoppedReason: ReasonEnumType [0..1]

+ NumberOfPhasesUsed: CounterType [0..1]

+Transaction + MaxAllowedHours: SecondsType [0..1]

+ MaxAllowedKwh: PowerType [0..1]

0..\* + TriggerReason: TriggerReasonEnumType [0..1]

+Transaction + RemoteStartID: RequestIdType [0..1]

+Transaction 0..1

«BusinessComponent»

**RootModel::ChargingStationType**

+ChargePoint 0..1

+ChargingProfile 0..\*

+ChargingProfile

0..1

+ChargingProfile 0..\*

«BusinessComponent»

**ChargingProfileType**

*IdentifiedObject*

{ConnectorUse}

+ Id: TransactionIdType [0..1]

0..\* + RunningCost: AmountType [0..1]

+ TotalCost: AmountType [0..1]

+Transaction 0..1

+EV 0..1

+ TransactionId: TransactionIdType [0..1]

«Content»

+ChargingProfile + StackLevel: CounterType [0..1]

+ Primary: IndicatorType [0..1] 0..\* + TimeBase: DateTimeType [0..1]

+ ChargingProfilePurpose: ChargingProfilePurposeEnumType [0..1]

+ ChargingProfileKind: ChargingProfileKindEnumType [0..1]

+Connector +Connectonre 0..1..1

0..\*+Con

ctor0

*DeviceType*

«BusinessComponent»

**RootModel::ConnectorType**

«Content»

+ ConnectorType: ConnectorEnumType [0..1]

+EV

+EV 0..1

0..1

«BusinessComponent»

**RootModel::EVType**

+EV 0..1

+EVCC 0..1

+EV 0..\*

+EV 0..1

+ChargingNeeds

0..1

+ RecurrencyKind: RecurrencyKindEnumType [0..1]

+ ValidFrom: DateTimeType [0..1]

+ ValidTo: DateTimeType [0..1]

+ ChargingLimitSource: ChargingLimitSourceEnumType [0..\*]

«BusinessComponent»

**ChargingNeedsType**

«Content»

+ RequestedEnergyTransfer: EnergyTransferModeEnumType

+ DepartureTime: DateTimeType [0..1]

[0..1]

+ CableCapacity: CurrentType [0..\*]

«BusinessComponent»

**RootModel::EVCCType**

+ChargingNeeds 0..\*

0..\* +ChargingNeeds

«Content»

+ EMAID: ChargingContractIdentifierType [0..1]

+ ContractSignatureEncryptedPrivateKey: SmallBinaryObjectType [0..1]

+ DHPublicKey: PublicKeyType [0..1]

+ Signature: SignatureType [0..1]

+ACChargingParameters 0..1

«BusinessComponent»

**ACChargingParametersType**

+ContractSignatureCertificateChain 0..1

*IdentifiedObject*

+RootCertificate

0..\*

«Content»

+ EnergyAmount: EnergyAmountType [0..1]

+ EVMinCurrent: CurrentType [0..1]

+ EVMaxCurrent: CurrentType [0..1]

+ EVMaxVoltage: VoltageType [0..1]

+SAProvisioningCertificateChain

0..1

«BusinessComponent»

**Security::CertificateChainType**

«Content»

]

+DCChargingParameters

«Content»

+ X509IssuerName: IssuerIdentifierType [0..1

+ X509SerialNumber: IntegerType [0..1]

«BusinessComponent»

**Security::X509IssuerSerialType**

|  |  |
| --- | --- |
|  | 0..1 |
| «BusinessComponent»  DCChargingParametersType | |
| «Content»  + EVMaxCurrent: NumericIdentifierType [0..1]  + EVMaxVoltage: NumericIdentifierType [0..1]  + EnergyAmount: NumericIdentifierType [0..1]  + EVMaxPower: NumericIdentifierType [0..1]  + StateOfCharge: PercentageType [0..1]  + EVEnergyCapacity: NumericIdentifierType [0..1]  + FullSoC: PercentageType [0..1]  + BulkSoC: PercentageType [0..1] | |

+CertificateChain + Certificate: Base64String800Type [0..1]

+ CertificateSerial: CI20TextType [0..1] 0..\* + ExpiryDate: DateType [0..1]

+OEMProvisioningCertificate *::IdentifiedObject*

+ MRID: NumericIdentifierType [0..1] 0..1 + ObjectID: IdentifierType [0..\*]

+ Name: NameType [0..1]

+ AliasName: NameType [0..\*]

+ Description: TextType [0..\*]

+ChildCertificate 0..\*

*Figure 15. OCPP Information Model: SmartCharging*

+家长证书 0..1

+计量功能

0..1

«PRIMSimpleType»

*(from PrimitiveDataTypes)*

«Supplementary»

+ measureUnit: String [0..1]

+ measureUnit.CodeListVersion: String [0.

«CDTComplexType,LDT,ECDMChange,...



«BusinessComponent»

**MeterValueType**

«Content»

+ Timestamp: DateTimeType [0..1]

.1]

+MeterValue 0..\*

+MeterValue 0..\*

+SampledValue 0..\*

*(from CoreDataTypes)*

«BusinessComponent»

**SampledValueType**

+MeteringFunction

0..1

«Content»

+ Value: MeasureType [0..1]

+ Context: ReadingContextEnumType [0..1]

+ Measurand: MeasurandEnumType [0..1]

+ Phase: PhaseEnumType [0..1]

+ Location: LocationEnumType [0..1]

+Meter 0..1

+Meter 0..1

*DeviceType*

«BusinessComponent»

**MeterType**

0..1

+SampledValue

«Content»

+ ClockAlignedDataInterval: SecondsType [0..1]

+ MeterValuesAlignedData: MeasurandEnumType [0..\*]

+ MeterValuesAlignedDataMaxLength: CounterType [0..1]

+ TxnStoppedAlignedData: MeasurandEnumType [0..\*]

+ TxnStoppedAlignedDataMaxLength: CounterType [0..1]

«Content»

+ Location: LocationEnumType [0..1]

+ SupportedMeasurands: MeasurandEnumType [0..\*]

+UnitOfMeasure 0..1

+SignedMeterValue + MeterValueSignature: Base64String2500Type [0..1]

+ SignatureMethod: SignatureMethodEnumType

0..1 + EncodingMethod: EncodingMethodEnumType [0..1]

+ EncodedMeterValue: Base64String512Type [0..1]

«BusinessComponent»

**UnitOfMeasureType**

+ Unit: CI20TextType [0..1]

+ Multiplier: IntegerType [0..1]

«BusinessComponent»

**ClockAlignedMeteringFunctionType**

«BusinessComponent»

**MeteringFunctionType**

*IdentifiedObject*

«BusinessComponent»

**RootModel::DeviceFunctionType**

*Figure 16. OCPP Information Model: Metering*

g

+Component 0..\*

+Variable 0..1

+Variable

1

+Variable 0..1

0..1 +Variable

+VariableMonitorin 1..\*

+VariableAttribute 0..\*

e

+VariableCharacteristics 0..\*

e

+ Units: MeasurementUnitType

+ DataType: DataEnumType

+ MinLimit: IntegerType [0..1]

+ MaxLimit: IntegerType [0..1]

+ ValuesList: CI500TextType [0..1]

+ SupportsMonitoring: IndicatorTyp

«BusinessComponent»

**VariableCharacteristicsType**

+ Type: AttributeEnumType [0..1] = Actual

+ Value: CI255TextType

+ Mutability: MutabilityEnumType [0..1]

+ Persistence: IndicatorType [0..1]

+ Constant: IndicatorType

«BusinessComponent»

**VariableAttributeType**

+ Value: MeasureType

+ Type: MonitorEnumType

+ Severity: SeverityEnumType

+ Cleared: IndicatorType [0..1] = fals

+ Transaction: IndicatorType [0..1]

«BusinessComponent»

**VariableMonitoringType**

+ Name: CI50TextType [0..1]

+ Instance: CI50TextType [0..1]

«BusinessComponent»

**VariableType**

+ Name: CI50TextType [0..1]

+ Instance: CI50TextType [0..1]

+ evse: NumericIdentifierType [0..1]

+ connector: NumericIdentifierType [0..1]

«BusinessComponent»

**ComponentType**

*Figure 17. OCPP Information Model: Device Model*



*DeviceType*

*IdentifiedObject*

«BusinessComponent»

**SecurityProfileType**

«BusinessComponent»

**RootModel::ChargingStationType**

«Content»

*::IdentifiedObject*

+ MRID: NumericIdentifierType [0..1]

+ ObjectID: IdentifierType [0..\*]

+ Name: NameType [0..1]

+ AliasName: NameType [0..\*]

+ Description: TextType [0..\*]

«deprecated, Content»

+ ChargeBoxSerialNumber: SerialNumberType [0..1]

«Content»

+ EVSECount: CounterType [0..1]

+ChargePoint 0..1

*IdentifiedObject*

«BusinessComponent»

**Components::LogType**

+SecurityLog

+SecurityProfile

0..1 + Filename: FilenameType [0..1]

+ RemoteLocation: URIType [0..1]

+ LogStartedDateTime: DateTimeType [0..1]

+ LogStoppedDateTime: DateTimeType

+ OldestTimestamp: DateTimeType [0..1]

+ LatestTimestamp: DateTimeType [0..1]

«Content»

0..1

+ManufacturerHierarchy +ChargePointOperatorHierarchy

0..1 0..1

«BusinessComponent»

**BasicAuthenticationProfileType**

«BusinessComponent»

**TLSDigestProfileType**

«BusinessComponent»

**CertificateHierarchyType**

«Content»

+ UserName: UserNameType [0..1]

+ Password: PasswordType [0..1]

«Content»

+ ChargePointCertificate: X509CertificateDigestType [0..1]

+ CentralServerCertificate: X509CertificateDigestType [0..1]

«Content»

+ Type: CertificateHierarchyCodeType [0..1]

+CertificateChain 0..\*

+CentralServerCertificate

0..1

+ChargePointCertificate

*IdentifiedObject*

«BusinessComponent»

**CertificateChainType**

+RootCertificate 0..\*

«BusinessComponent»

**TLSBasicAuthenticationProfileType**

«Content»

0..1 + Certificate: Base64String800Type [0..1]

«BusinessComponent»

**X509IssuerSerialType**

+CentralServerCertificate + CertificateSerial: CI20TextType [0..1]

+ ExpiryDate: DateType [0..1]

0..1

+FirmwareSigningCertificate

0..1

+RootCertificate

0..1 + X509IssuerName: IssuerIdentifierType [0..1]

+ X509SerialNumber: IntegerType [0..1]

«Content»

*IdentifiedObject*

«BusinessComponent»

**Components::FirmwareType**

+ParentCertificate 0..1

«Content»

+ Version: VersionType [0..1]

+ Status: FirmwareStatusEnumType [0..1]

+ Location: URIType [0..1]

+ RetrieveDateTime: DateTimeType [0..1]

+ InstallDateTime: DateTimeType [0..1]

+ Signature: Base64String800Type [0..1]

+ ServiceURL: URIType [0..1]

+ChildCertificate 0..\*

*Figure 18. OCPP Information Model: Security-Profiles*

..1]



«BusinessComponent»

**Components::IdentifiedObject**

«Content»

+ MRID: NumericIdentifierType [0..1]

+ ObjectID: IdentifierType [0..\*]

+ Name: NameType [0..1]

+ AliasName: NameType [0..\*]

+ Description: TextType [0..\*]

«BusinessComponent»

**RootModel::DeviceFunctionType**

«BusinessComponent»

**OCSPRequestDataType**

+AuthorizationList

0..\*

+ HashAlgorithm: HashAlgorithmEnumType [0..1]

+ IssuerNameHash: IssuerNameType [0..1]

+ IssuerKeyHash: HashType [0..1]

+ SerialNumber: SerialNumberType [0..1]

+ ResponderURL: URIType [0..1]

+OCSPCertificateHash 0..\*

«Content»

+ VersionNumber: IntegerType [0..1] +AuthorizationCache

+ MaxLength: CounterType [0..1]

+ MaxTransmitCount: CounterType [0..1]

0..1

+ LocalAuthListEnabled: IndicatorType [0..1]

+ LocalAuthListMaxLength: CounterType [0..1]

+ SendLocalListMaxLength: CounterType [0..1]

«BusinessComponent»

**AuthorizationFunctionType**

+AuthorizationFunction

+AuthorizationList 0..1

0..\*

«Content»

+ AuthorizationListSupported: IndicatorType [0..1]

+ LocalAuthorizationEnabled: IndicatorType [0..1]

+ LocalAuthorizeOffline: IndicatorType [0..1] +Transaction

+ LocalPreAuthorize: IndicatorType [0..1]

+ AuthorizationCacheEnabled: IndicatorType [0..1]

0..\*

+ AllowOfflineTxForUnknownID: IndicatorType [0..1]+AuthorizationFunction

+ AuthorizeRequestTxRequests: IndicatorType [0 0..1

+ LawEnforcementGroupId: TokenType [0..1]

+LocalAuthorization

«BusinessComponent»

**IDTokenType**

+IDToken

+Transaction 0..1

0..\* + IDToken: TokenType [0..1]

0..\*

+ Type: IDTokenEnumType [0..1]

+IDToken 0..1

+IDTokenInfo 0..1

«BusinessComponent»

**IDTokenInfoType**

«Content»

+ Status: AuthorizationStatusEnumType [0..1]

+ CacheExpiryDateTime: DateTimeType [0..1]

+ Language1: LanguageType [0..1]

+ Language2: LanguageType [0..1]

+ ChargingPriority: NumericIdentifierType [0..1]

+ ListID: IdentifierType [0..1]

+ ListEntryTimestamp: DateTimeType [0..1]

+IDTokenInfo +TariffMessage

«BusinessComponent»

**Components::MessageContentType**

0..\*

0..1

«Content»

+IDTokenInf +PersonalMessage

+ Format: MessageFormatEnumType [0..1]

+ Language: LanguageType [0..1]

0..\*

0..1 + Content: MessageType [0..1]

«Content»

+ SeqNo: CounterType [0..1]

+ ChargingState: ChargingStateEnumType [0..1]

+ Offline: IndicatorType [0..1]

+ StartedTimestamp: DateTimeType [0..1]

+ StoppedTimestamp: DateTimeType [0..1]

+ TimeSpentCharging: SecondsType [0..1]

+ StoppedReason: ReasonEnumType [0..1]

+ NumberOfPhasesUsed: CounterType [0..1]

+ MaxAllowedHours: SecondsType [0..1]

+ MaxAllowedKwh: PowerType [0..1]

+ TriggerReason: TriggerReasonEnumType [0..1]

+ RemoteStartID: RequestIdType [0..1]

+ Id: TransactionIdType [0..1]

+ RunningCost: AmountType [0..1]

+ TotalCost: AmountType [0..1]

«BusinessComponent»

**Transactions::TransactionType**

«BusinessComponent»

**AuthorizationListType**

*Figure 19. OCPP Information Model: Security-Authorization*

o

«enum, Facet» Central eMAID ISO14443 ISO15693

Local NoAuthorization KeyCode

« enumeration,BDT...

**CodeLists:: IDTokenEnumType**

« enumeration,BDTEnu...

**CodeLists:: AuthorizationStatusEnumType**

«enum, Facet» Accepted Blocked Expired Invalid NoCredit

NotAllowedTypeEVSE NotAtThisLocation NotAtThisTime ConcurrentTx Unknown



«BusinessComponent»

**Components::IdentifiedObject**

«Content»

+ MRID: NumericIdentifierType [0..1]

+ ObjectID: IdentifierType [0..\*]

+ Name: NameType [0..1]

+ AliasName: NameType [0..\*]

+ Description: TextType [0..\*]

«BusinessComponent»

**VPNType**

«Content»

+ Server: URIType [0..1]

+ User: UserNameType [0..1]

+ Group: GroupNameType [0..1]

+ Password: PasswordType [0..1]

+ Key: VPNKeyType [0..1]

+ Type: VPNCodeType [0..1]

«BusinessComponent»

**RootModel::DeviceFunctionType**

+VPN 0..1

«BusinessComponent»

**NetworkConnectionProfileType**

«Content»

+ Priority: IntegerType [0..1]

+ HeartbeatInterval: SecondsType [0..1]

+ OcppVersion: OCPPVersionCodeType [0..1]

+Receiver

+Trigger

0..\*

0..\*

«Content»

+ TriggerCode: MessageTriggerEnumType [0..1]

+ OcppTransport: OCPPTransportCodeType [0..1]

«BusinessComponent» + OcppCsmsUrl: URIType [0..\*]

**APNType** + OcppInterface: OCPPInterfaceCodeType [0..1]

«Content»

+ APN: URIType [0..1]

+ APNUserName: UserNameType [0..1]

+ APNPassword: PasswordType [0..1]

+ SimPin: PINEnumType [0..1]

+ PreferredNetwork: MobileNetworkIDType [0..1]

+ UseOnlyPreferredNetwork: IndicatorType [0..1]

+ APNAuthentication: APNAuthenticationCodeType [0..1]

+CommunicationFunction 0..\*

+APN 0..1

+CommunicationModule 0..1

«BusinessComponent»

**ModemType**

«Content»

+ ICCID: SimIdentifierType [0..1]

+ IMSI: SimIdentifierType [0..1]

«BusinessComponent»

**TriggerType**

图 20. OCPP 信息 模型： 通信

«Content»

+ MRID: NumericIdentifierType [0..1]

+ ObjectID: IdentifierType [0..\*]

+ Name: NameType [0..1]

+ AliasName: NameType [0..\*]

+ Description: TextType [0..\*]

«BusinessComponent»

**Components::IdentifiedObject**

«BusinessComponent»

**SAScheduleType**

+SASchedule 0..\*

«Content»

«BusinessComponent»

**SalesTariffType**

+SalesTariff 0..1

+ SalesTariffDescription: TariffDescriptionType [0..1]

+ NumEPriceLevels: CounterType [0..1]

0..1

+SalesTariffEntry 1..1024

.1]

«Content»

+ PMax: PowerType [0..1]

«BusinessComponent»

**PMaxScheduleType**

+ EPriceLevel: UnsignedIntegerType [0.

«BusinessComponent»

**SalesTariffEntryType**

0..\* 0..1

+ConsumptionCost

+Cost

0..3 +RelativeTimeInterval 0..1

+RelativeTimeInterval

«Content»

+ StartValue: NumericType [0..1]

«BusinessComponent»

**ConsumptionCostType**

|  |  |
| --- | --- |
|  |  |
| «BusinessComponent»  RelativeTimeIntervalType | |
| «Content»  + Start: SecondsType [0..1]  + Duration: SecondsType [0..1] | |

0..1

0..\*



«Content»

+ CostKind: CostKindCodeType [0..1]

+ Amount: AmountType [0..1]

+ AmountMultiplier: IntegerType [0..1]

«BusinessComponent»

**CostType**

*Figure 21. OCPP Information Model: SecondaryActorSchedule*