

Open Clearing House Protocol (OCHP)



Version	0.1
Status	Final
Date	29 March 2012

Version Log	Version	Date	Author	Commentary
	Concept	28-02-2012	Andreas Pfeiffer/Christian Pestel, smartlab	Functional specification
	First draft	23-03-12	H.J. Voltman, Liandon	
	Second draft	26-03-12	I. van Gemert, Liandon	
	Final	29-03-2012	Andreas Pfeiffer/Christian Pestel, smartlab	

The concept has been agreed upon by:	
Peter Buycks	enovates
Arjan Wargers	Stichting e-laad
Raymund Rutten	Stichting e-laad
Andreas Pfeiffer	smartlab Innovationsgesellschaft mbH
Christian Pestel	smartlab Innovationsgesellschaft mbH

Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages

Disclaimer

© smartlab, bluecorner.be, e-laad.nl 2012

All rights reserved. This document is protected by international copyright law and may not be reprinted, reproduced, copied or utilized in whole or in part by any means including electronic, mechanical, or other means without the prior written consent of smartlab.

Abbreviations

Abbreviation	Meaning
CDR	Charge Data Record
CHS	Clearing House System
CMS	Chargepoint Management System
eCHS	European Clearing House System
EV	Electrical Vehicle
EVCO-ID	Electrical Vehicle Contract Identifier
EVSE-ID	Electrical Vehicle Supply Equipment Identifier
ID	Identifier
PDU	Protocol Data Unit

Content

1	Introduction	5
1.1	PRIMARY STAKEHOLDERS ELECTRICAL VEHICLES	5
1.1.1	EV-customer	6
1.1.2	EV-Service Provider	6
1.1.3	Local Charge Provider	6
1.1.4	Charge Pole owner	6
1.1.5	Energy Supplier	6
1.1.6	Grid operator	6
1.2	EV CLEARING HOUSE	7
2	CHS-CMS Interface description	7
2.1	AUTHENTICATE	8
2.1.1	System authorization by eCHS	8
2.1.2	Partner Management	8
2.2	EXCHANGE AUTHORISATION DATA	8
2.2.1	Upload own authorisation data (roaming list) to the eCHS	8
2.2.2	Download global roaming authorisation data from eCHS	8
2.3	EXCHANGE CHARGE DATA	8
2.3.1	Upload Charge Data Records	8
2.3.2	Clear Roaming Charge Data Records	9
2.3.3	Download Extracted Roaming Charge Data Records	9
3	Messages	10
3.1	AUTHENTICATE.REQ	10
3.2	AUTHENTICATE.CONF	10
3.3	ADDCDRs.REQ	10
3.4	ADDCDRs.CONF	10
3.5	GETCDRs.REQ	11
3.6	GETCDRs.CONF	11
3.7	CLEARCDRs.REQ	11
3.8	CLEARCDRs.CONF	11
3.9	GETROAMINGAUTHORISATIONLIST.REQ	11
3.10	GETROAMINGAUTHORISATIONLIST.CONF	11
3.11	SETROAMINGAUTHORISATIONLIST.REQ	12
3.12	SETROAMINGAUTHORISATIONLIST.CONF	12
4	Types	13
4.1	AUTHRESULTTYPE	13
4.2	CDRINFO	13
4.3	RESULT	14
4.4	ROAMINGAUTHORISATIONINFO	14
4.5	INFRASTRUCTUREPROVIDERTYPE	15
4.6	TOKENTYPE	15
4.7	MANDATORYTYPE	15
4.8	ACTIVATEDTYPE	15
5	Binding to Transport Protocol	16
5.1	USER IDENTITY HEADER	16

1 Introduction

1.1 Primary Stakeholders Electrical Vehicles

To understand the playfield involved in this eCHS figure 1 illustrates the stakeholders according to the Dutch e-mobility market. As every country has different local stakeholders the model is to be further developed.

Stakeholder	Example
Grid Operator	Liander
EV-Customer	
Energy Supplier	Essent, Nuon
EV-Service Provider	E-Laad depicted.
Local Charge Provider	Q-Park, IKEA, Arena
Chargepole Owner	City of Amsterdam

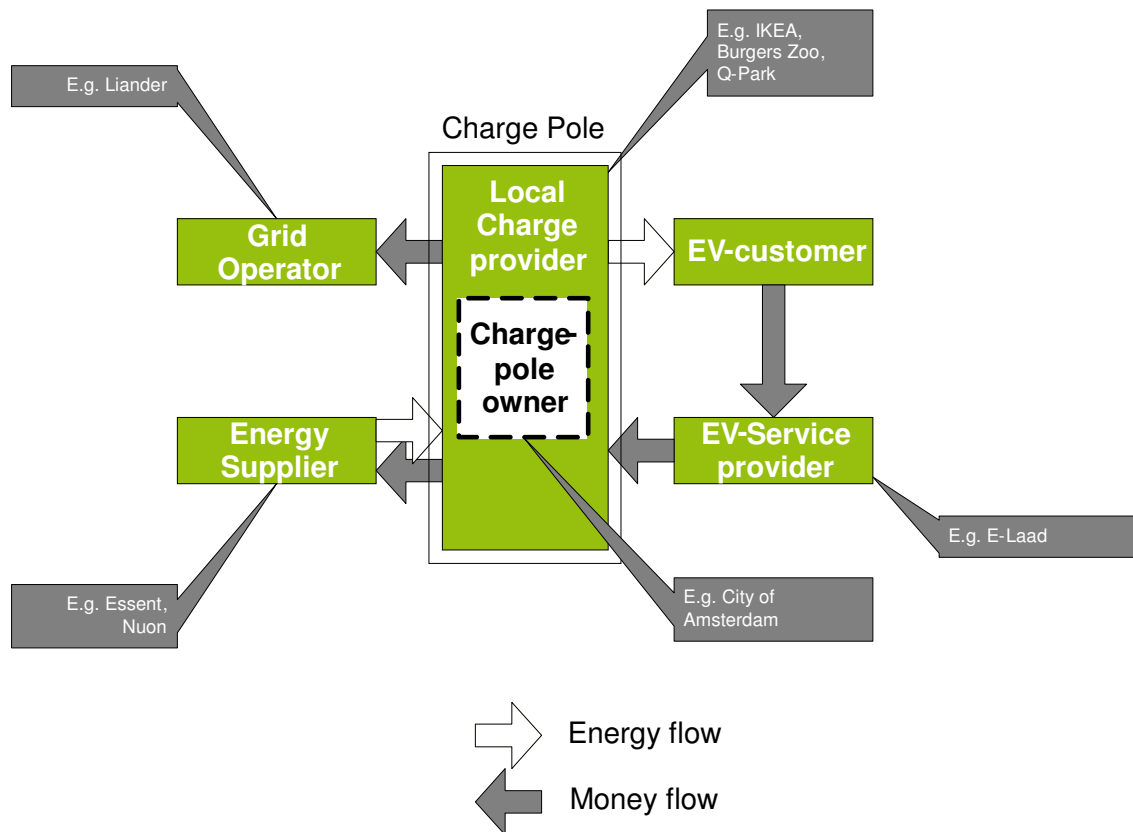


Figure 1 Relations between EV-stakeholders

1.1.1 EV-customer

An EV-customer uses a electric vehicle for local transport and needs electricity for direct mobility needs. The EV-customer is able to charge his/her car at home or uses public infrastructure. More and more EV-chargepoles are available at public accessible locations owned by the local government. Also EV-charging locations are available in for example retail locations like IKEA. In this specific case IKEA operates like a local charge provider.

1.1.2 EV-Service Provider

The EV-Service Provider operates as a contract party for the EV-customer. The EV-Service Provider takes care of the enduser authentication and billing processes. The EV-Service Provider provides the EV-customer an RFID identification card that gives authorisation to use the contracted Charge poles.

To fulfil its role, the EV-Service Provider makes use of a Chargepoint Management System (CMS) that manages charging stations and its chargepoints and has the information for authorizing users for using its chargepoints.

1.1.3 Local Charge Provider

The (local) Charge provider operates as contract party for the Energy Supplier. The Charge Provider purchases the Electricity from the Energy Supplier and will send the EV-service provider an invoice for the delivered Electricity.

1.1.4 Charge Pole owner

Sometimes the (local) Charge Provider is not the same party as the owner of the Charge Pole. For instance, the City of Amsterdam can be owner of a Charge Pole, located in a private Parking Place. For that specific case, the Parking company operates as Charge provider, while the City of Amsterdam is owner of the physical Charge Pole.

1.1.5 Energy Supplier

The Energy Supplier is contracted by the (local) Charge provider to deliver the Electricity against the contracted tariffs.

1.1.6 Grid operator

The Grid operator is owner of the public Electricity network and is responsible for the connection of the Charge pole to the grid and for the transport of the Electricity.

1.2 EV Clearing House

The positioning of the EV Clearing House is made more clear in figure 2.

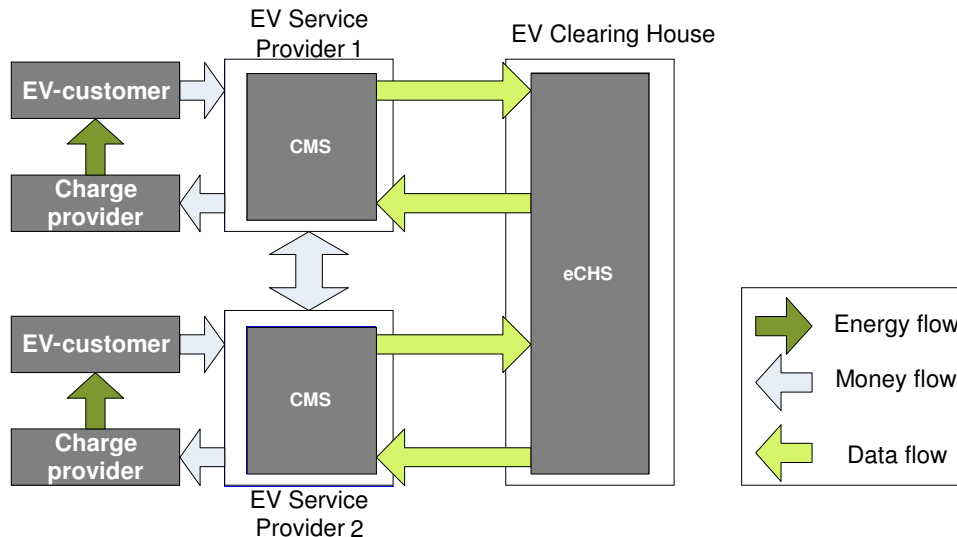


Figure 2: Positioning EV Clearing House

To describe the principal processes, two different EV Service Providers are depicted in figure 2, but the processes can easily be expanded to more (international) EV Service Providers.

The basic idea is that a Clearing House gives “Roaming Support” for every EV Service Provider. The ultimately goal is that the EV-customer can easily charge his Electrical Vehicle on every Charge Pole of every EV Service Provider. With Roaming Support, provided by the Clearing House, the complexity of relationships can be reduced: from many-to-many bilateral EV Service Provider relationships towards a one-to-many relationship between Clearing House and the EV Service Providers.

2 CHS-CMS Interface description

The interfaces between the European Clearing House System (eCHS) of the Clearing House and the Charge point Management Systems (CMS) of the different EV Service Providers consist mainly out of the two following components:

1. Exchange of Authorisation Data
2. Exchange of Charge Data

The exchange of data will take place via standardized web services.

An essential basic rule is that the lead for partner management will remain at each individual CMS. This rule has the following consequences for each CMS:

- A client for every roaming partner is implemented in each CMS (identification through provider ID);
- Data will only be accepted if the provider ID of the partner is known at CMS.

2.1 Authenticate

2.1.1 System authorization by eCHS

Prior to all interactions, in the eCHS system a user_id and a password has to be defined for each EV Service Provider that needs access to the eCHS. That user_id and password has to be handed over from Clearing House towards the EV Service Provider. From that point, an EV Service Provider can access the eCHS.

The first step in exchanging data is that a CMS user gets authorized access to eCHS. This authorised access can be received in the following way.

1. CMS sends an Authenticate.req PDU.
2. eCHS responds with an Authenticate.conf PDU.

Note that when the authentication was successful, the Authenticate.conf PDU contains the token that MUST be used in the soap header of all other requests from CMS to eCHS.

2.1.2 Partner Management

New eCHS users are created setting user ID and password. These administrator tasks are not part of the CMS – eCHS interface and are therefore not described in this document

2.2 Exchange Authorisation Data

2.2.1 Upload own authorisation data (roaming list) to the eCHS

Each CMS has to exchange the own EV-Customer authorisation data with the Clearing House, to share that data with other EV Service providers.

The upload of the own roaming list is done in the following way:

1. CMS sends the SetRoamingAuthorisationList.req PDU.
2. eCHS responds with a SetRoamingAuthorisationList.conf PDU.

2.2.2 Download global roaming authorisation data from eCHS

A CMS downloads the global authorisation data repository from the eCHS.

The download of the global roaming list is done in the following way:

1. CMS sends the GetRoamingAuthorisationList.req PDU.
2. eCHS responds with GetRoamingAuthorisationList.conf PDU.

2.3 Exchange Charge Data

2.3.1 Upload Charge Data Records

Local roaming charge data records are sent from the CMS to the eCHS.

The upload has to be done in the following way:

1. CMS sends the AddCDRs.req PDU.
2. eCHS responds with: AddCDRs.conf PDU.

2.3.2 Clear Roaming Charge Data Records

The clearing of all pending roaming CDRs can be done in the following way:

1. CMS sends ClearCDRs.req PDU.
2. eCHS responds with a ClearCDRs.conf PDU.

Note that the CMS must be added to the administrator group to be allowed to perform this operation.

2.3.3 Download Extracted Roaming Charge Data Records

Cleared roaming charge data records, held at the eCHS are sent from the eCHS to the CMS. Note that only cleared CDRs that concern the CMS are sent.

The download has to be done in the following way:

1. CMS sends GetCDRs.req PDU.
2. eCHS responds with a GetCDRs.conf PDU.

3 Messages

3.1 Authenticate.req

This contains the field definition of the Authenticate.req sent by the an eCHS partner to the eCHS.

Field Name	Field Type	Card.	Description
userId	string	1..1	Mandatory. This contains the user identification
Password	string	1..1	Mandatory. This contains the user's password.

3.2 Authenticate.conf

This contains the field definition of the Authenticate.conf sent by the the eCHS as reponse to the Authenticate.req.

Field Name	Field Type	Card.	Description
resultCode	AuthResultType	1..1	Mandatory. This contains the authentication result.
resultDescription	string	1..1	Mandatory. This contains the reason for failure.
authToken	string	1..1	Mandatory This contains the authToken that should be supplied (in the soap header) to all the other requests.

3.3 AddCDRs.req

This contains the field definition of the AddCDRs.req sent by an eCHS partner to the eCHS.

Field Name	Field Type	Card.	Description
cdriInfoArray	CDRIInfo	1..*	Mandatory. This contains one or more Charge Data Records.

3.4 AddCDRs.conf

This contains the field definition of the AddCDRs.conf sent by the the eCHS as reponse to the AddCDRs.req.

Field Name	Field Type	Card.	Description
result	Result	1..1	Mandatory. This contains the result of the AddCDRs.req.

3.5 GetCDRs.req

This contains the field definition of the GetCDRs.req sent by the an eCHS partner to the eCHS.

No fields are defined.

3.6 GetCDRs.conf

This contains the field definition of the GetCDRs.conf sent by the the eCHS as reponse to the GetCDRs.req.

Field Name	Field Type	Card.	Description
result	Result	1..1	Mandatory. This contains the result of the GetCDRs.req.
docLink	string	1..1	Mandatory. This contains the link to the cleared CDR pdf document.
cdrInfoArray	CDRInfo	0..*	Optional. This contains the CDR's that have been cleared (in the last call to ClearCDRs).

3.7 ClearCDRs.req

This contains the field definition of the ClearCDRs.req sent by the an eCHS partner to the eCHS.

No fields are defined.

3.8 ClearCDRs.conf

This contains the field definition of the ClearCDRs.conf sent by the the eCHS as reponse to the ClearCDRs.req.

Field Name	Field Type	Card.	Description
result	Result	1..1	Mandatory. This contains the result of the ClearCDRs.req.

3.9 GetRoamingAuthorisationList.req

This contains the field definition of the GetRoamingAuthorisationList.req sent by the an eCHS partner to the eCHS.

No fields are defined.

3.10 GetRoamingAuthorisationList.conf

This contains the field definition of the GetRoamingAuthorisationList.conf sent by the the eCHS as reponse to the GetRoamingAuthorisationList.req.

Field Name	Field Type	Card.	Description
------------	------------	-------	-------------

Field Name	Field Type	Card.	Description
result	Result	1..1	Mandatory. This contains the result of the GetRoamingAuthorisationList.req.
roamingAuthorisationInfoArray	RoamingAuthorisationInfo	0..*	Optional. This contains the roaming authorisation records.

3.11 SetRoamingAuthorisationList.req

This contains the field definition of the SetRoamingAuthorisationList.req sent by the an eCHS partner to the eCHS.

Field Name	Field Type	Card.	Description
roamingAuthorisationInfoArray	RoamingAuthorisationInfo	1..*	Mandatory. This contains the roaming authorisation records.

3.12 SetRoamingAuthorisationList.conf

This contains the field definition of the SetRoamingAuthorisationList.conf sent by the the eCHS as reponse to the SetRoamingAuthorisationList.req.

Field Name	Field Type	Card.	Description
result	Result	1..1	Mandatory. This contains the result of the SetRoamingAuthorisationList.req.

4 Types

4.1 AuthResultType

Enumeration

ResultCode of the Authenticate.req.

Field Name	Field Type	Description
Accepted		User is authenticated
Denied		Used is not authenticated

4.2 CDRInfo

Class

Contains all information concerning a Charge Data Record

Field Name	Field Type	Description
cdrlId	string	Mandatory. Charge Data Record number. Unique per Infra_Provider_ID.
startDatetime	string	Mandatory. Start date and time of the charge session (logon with the RFID badge). Local time is used. Format is according to ISO8601 UTC+Offset. Example: "2011-06-01T11:45:30+02:00"
endDatetime	string	Mandatory. Start date and time of the charge session (logon with the RFID badge). Local time is used. Format is according to ISO8601 UTC+Offset. Example: "2011-06-01T11:45:30+02:00"
duration	string	Optional. Duration of the charge session. Example: "0:00:28"
volume	string	Mandatory. The volume delivered during the charge session in kWh. Four digits decimal precision (N4,N4). Example: "0,0004"
chargePointAddress	string	Optional. Address (street and possibly housenumber) of the location of the chargepoint.
chargePointZip	string	Optional. Where available the ZIP code of the location of the chargepoint, conforming to country standard
chargePointCity	string	City of the location of the chargepoint.
chargePointCountry	string	Optional. Country of the location of the chargepoint. Format is according to the three-character ISO-3166 code.
chargePointType	string	Optional. Charge Point Type.
productType	string	Mandatory. Identifies the type of the product that is delivered on the chargepoint.

Field Name	Field Type	Description
tariffType	string	Mandatory.
authenticationId	string	Optional. Customer identification, RFID decoded as stored in the central register.
evcold	string	Mandatory. Customer contract identification or ExternalID of the ChargeBadge as stored in the central register (as ExternalID). Example: "NL-ESS-123456-7".
meterid	string	Mandatory. Meter identification.
obisCode	string	Optional. OBIS object identification of the register in the meter. Format is according to the OBIS if the COSEM application layer (defined in IEC 62056-61) and as such aligned with the Dutch Smart Meter Requirements. Example: "1-1:1.8.0" is the 'Total' register (".0") of the 1st ("1-1") meter regarding power delivered 'from the grid' ("1.8").
chargePointId	string	Mandatory. Charge Point Identification. The Charge Point ID identifies the physical socket (within the Chargepoint Operator) that was used for charging in for the particular CDR.
serviceProviderId	string	Mandatory. Service Provider Identification (ie the recipient of the CDR), as stored in the central register
infraProviderId	string	Mandatory. Infra provider Identification (ie the sender of the CDR), as stored in the central register. This corresponds to the National_Hub_ID (Example: "e-laad", "Ladennetz", "Blue Corner").
evseld	string	Optional.

4.3 Result

Class

Contains result information.

Field Name	Field Type	Description
resultCode	int	Mandatory. 0 = ok, any other value = error
resultDescription	string	Mandatory. The reason of the error.

4.4 RoamingAuthorisationInfo

Class

Contains information about a roaming authorisation (card/token)

Field Name	Field Type	Description
evcold	string	Mandatory. Electrical Vehicle Contract Identifier

Field Name	Field Type	Description
roamingHubId	InfraStructureProviderType	Mandatory.
tokenType	TokenType	Mandatory.
tokenId	string	Mandatory.
printedNumber	string	Mandatory.
expiryDate	string	Mandatory. dd/mm/yyyy
pin	string	Mandatory.
pinMandatory	MandatoryType	Mandatory.
tokenActivated	ActivatedType	Mandatory.
hash	String	Mandatory.

4.5 InfraStructureProviderType

Enumeration

Field Name	Field Type	Description
Other		0
BlueCorner		1
E-laad		2
Ladenetz		3

4.6 TokenType

Enumeration

Field Name	Field Type	Description
Other		0
MifareCL		1
MifareDES		2
MSISDN		3

4.7 MandatoryType

Enumeration

Field Name	Field Type	Description
Not Mandatory		0
Mandatory		1

4.8 ActivatedType

Enumeration

Field Name	Field Type	Description
Deactivated		0
Activated		1

5 Binding to Transport Protocol

This section describes how the eCHS PDUs can be conveyed over SOAP. The rationale behind using SOAP as a transport is that SOAP already provides the infrastructure of sending messages. SOAP has a good support in the industry, which results in tools that improve the ease of implementing the protocol. The used version of SOAP MUST be 1.2.

5.1 User Identity Header

All requests to the eCHS (except Authenticate.req) MUST send in each request PDU, its Authentication token (received in the Authenticate.resp) in the SOAP header. The header name is “*UserIdentityHeader*” of XSD type “string”. For example:

```
<!-- Header containing the user identity token -->  
<ns:UserIdentityHeader>4b22f1316a88003846207a887d9f8d72</ns: UserIdentityHeader >
```