HUBJECT

Open intercharge Protocol

for Emobility Service Providers

VERSION 2.2 / MARCH 2018

Table of Contents

Table of Contents

Table of Contents

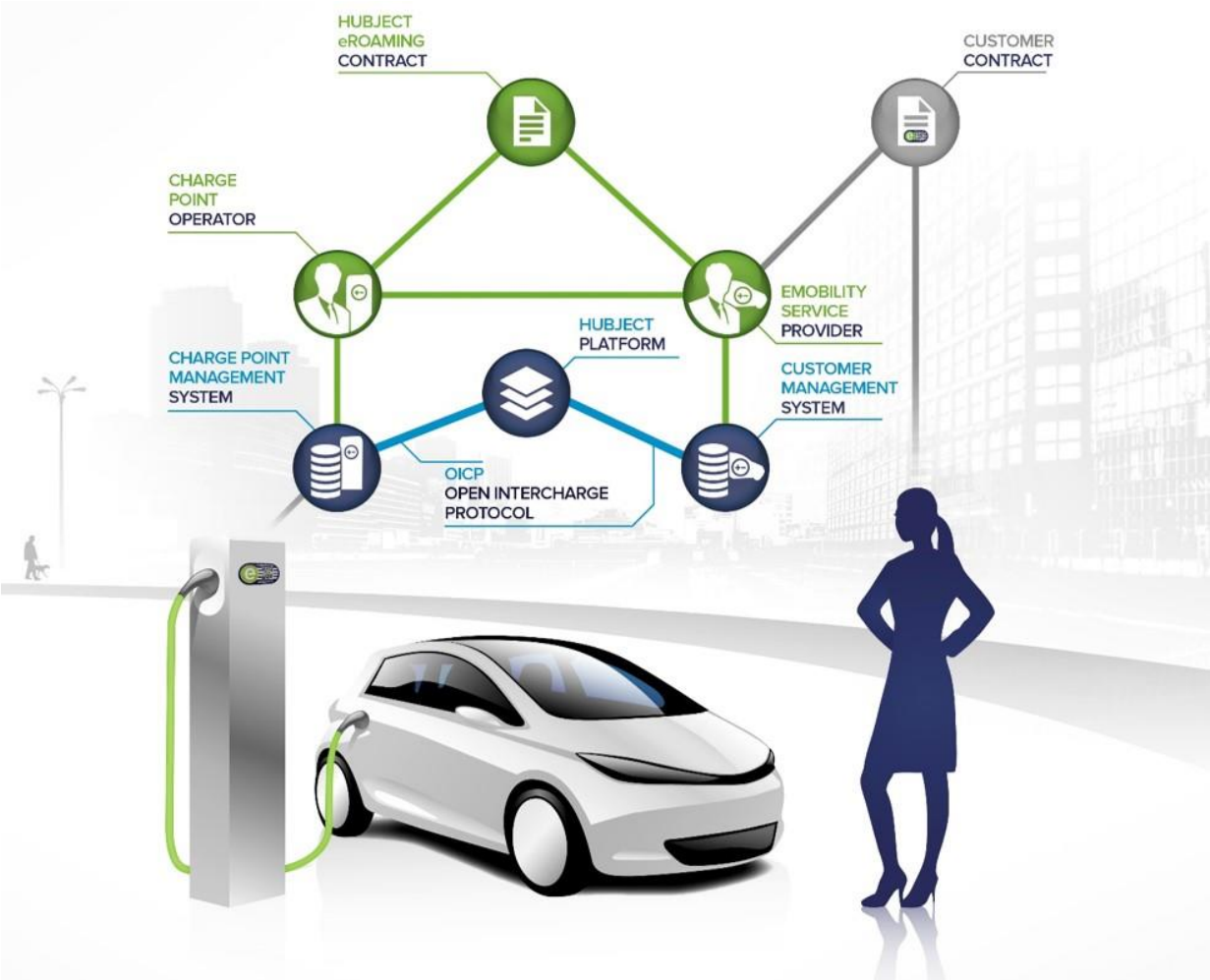
Table of Contents

Table of Contents

1 Introduction

1.1 The Hubject Platform

The goal of the “Hubject B2B Service Platform (HBS)” is to enable the electric mobility market by providing an information and transactional gateway for businesses such as charging infrastructure providers, mobility service providers and vehicle manufacturers.



The enabler functions of the platform include:

- Ensuring the interoperability of the public and semi-public infrastructure through promotion of accepted standards within the network and open business user interfaces to the platform
- Simplification of authentication and authorization procedures through a trustworthy instance as well as safekeeping of sensitive data through the uncoupling of personal data and anonymous user data
- Automation of contract-based business relationships between power suppliers, car manufacturers, infrastructure service providers as well as further mobility business parties
- B2B information services for the realization of advanced services within the areas of energy management, traffic management, vehicle reservations, intelligent charging, car sharing and intermodal mobility

1.2 The Emobility Service Provider (EMP)

As an EMP you profit from our solution CONNECT. It allows you to offer EV drivers access to public charging infrastructure across national borders. Your customers will be able to identify any available charging point through our easily recognizable compatibility symbol, the intercharge logo. By getting connected to the intercharge network your customers will be able to use every charge point of all participating CPOs via eRoaming.

For this purpose, you need to be onboarded to the eRoaming platform HBS (Hubject Brokering System). The HBS functions as an open emobility market place, which creates an open synergetic network that everyone profits from in the end.

In general, there are two different possibilities to be connected to the Hubject platform as an EMP.

Offline EMP:
The so called offline EMP has no real-time connection for authorization to the Hubject platform. This means that authentication data for the authorization MUST be sent and stored on the Hubject platform via the eRoamingPushAuthenticationData (see 4.1.1). The HBS will authorize charging sessions locally on the platform. Nevertheless, there can be a real-time connection for pulling dynamic POI data (see 4.1.3). Furthermore, the CDRs, resulting from each charging session, will not be directly forwarded to the EMP but will also be stored on the Hubject system. The EMP can then download these CDRs on demand via the eRoamingGetChargeDetailRecords service (see 4.1.6).

Online EMP:
The so called online EMP is fully connected to the Hubject platform via a real-time interface. This means that authorization requests are forwarded to the EMP's system in real-time. To ensure the online authorization, the EMP MUST implement the holistic authorization web service (see 0). Furthermore, the CDRs created for each charging session will be directly forwarded to the EMP's system. Therefore, online EMP's MUST implement the eRoamingChargeDetailRecord service (see 4.2.1).

1.3 Scope

The information exchange between Hubject and Electric Mobility Provider (EMP) systems or Charge Point Operator (CPO) systems is entirely based on web service communication. This document describes the corresponding service interfaces. The Open InterCharge Protocol (OICP) is the most widely implemented communication standard between EMP and CPO systems.

The information exchange is in most cases based on contractual relationships between EMPs and CPOs. In these cases, Hubject only processes service requests in case there is a valid contract for the requested service. How EMPs and CPOs manage their service contracts is out of the scope of this document because the contract management aspects of the platform are conducted via a GUI-based system component.

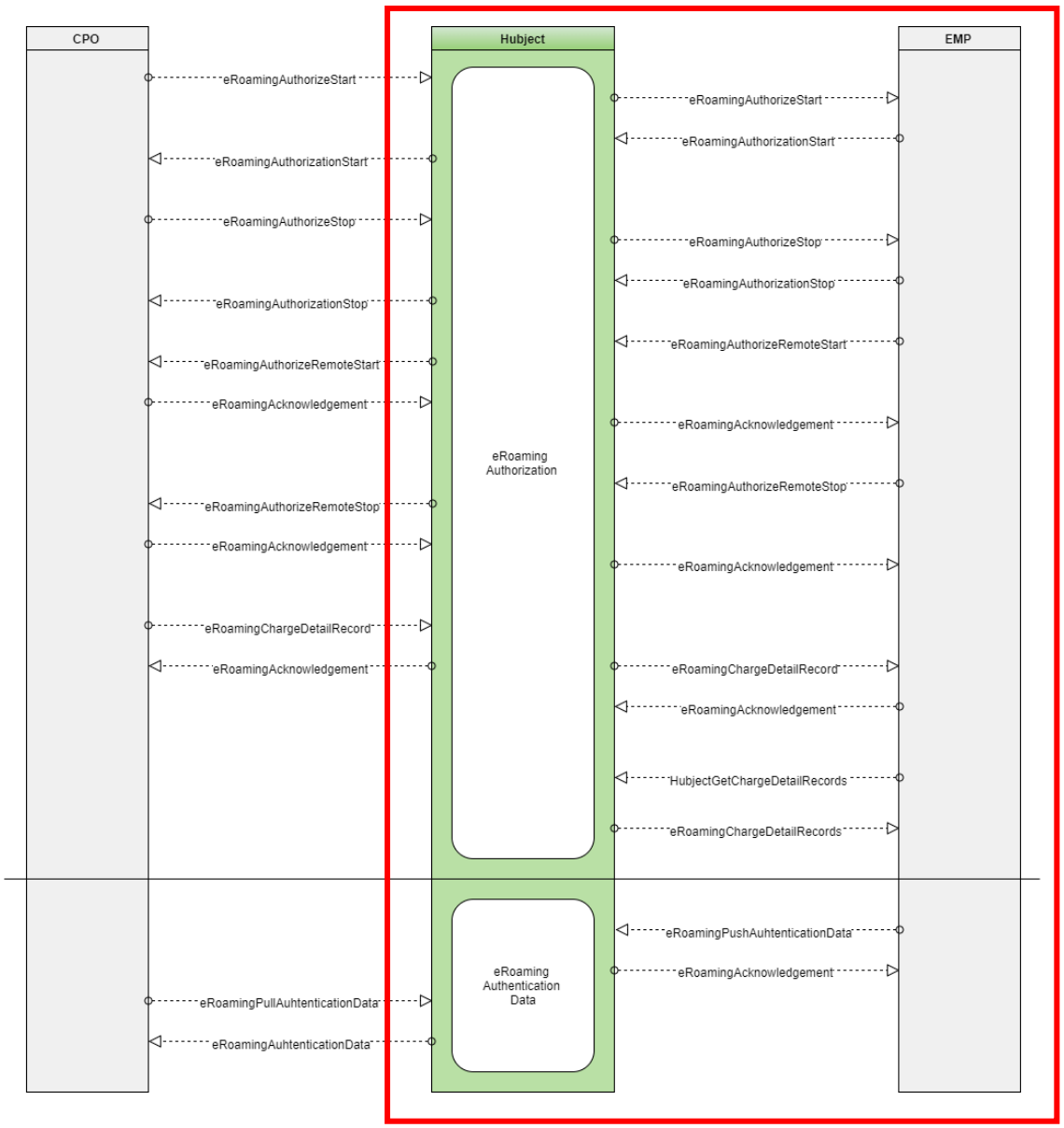
1.4 Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119].

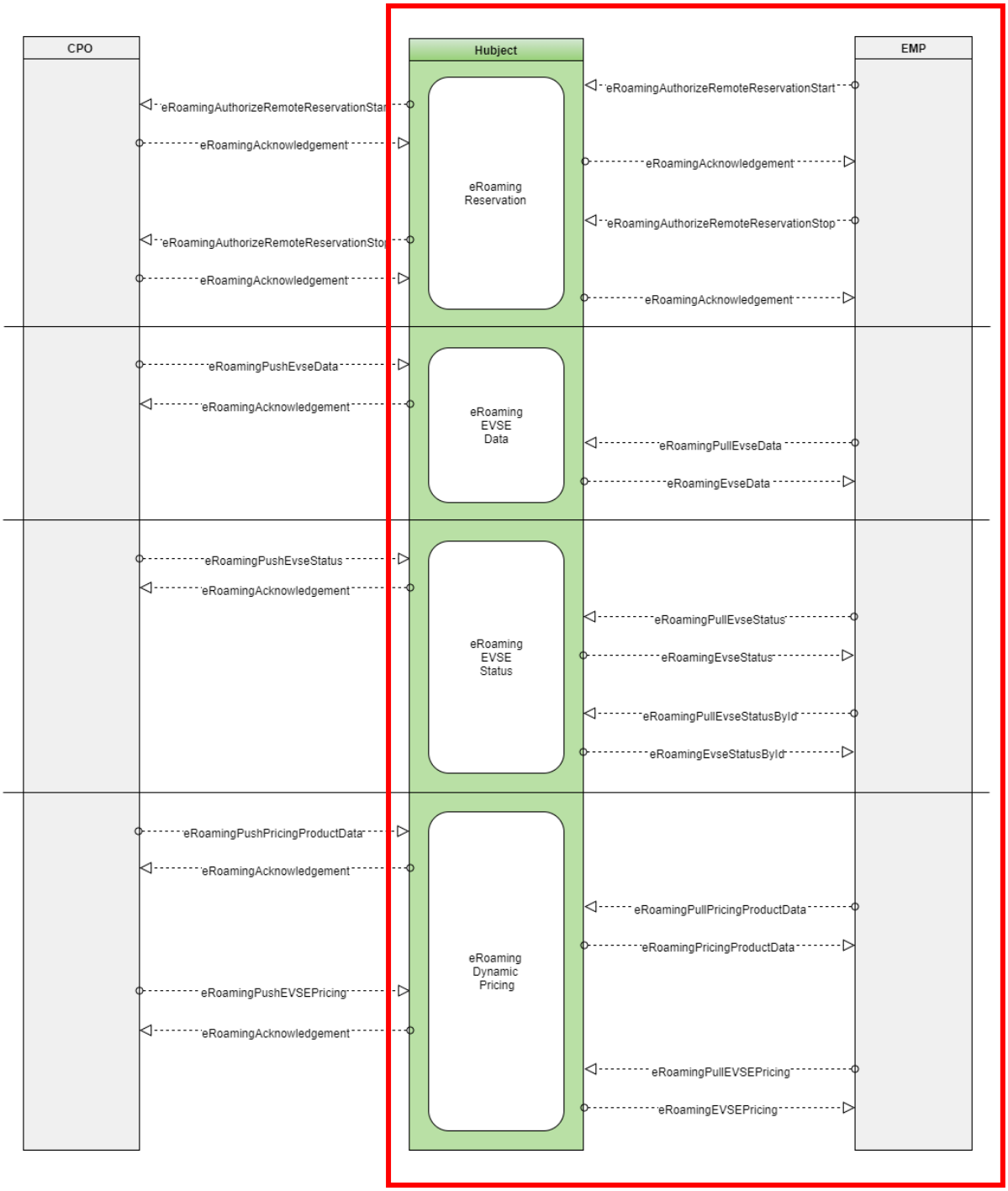
1.5 Overview

The following diagram gives an overview of all service operation messages that can be exchanged between Hubject and the corresponding EMP, respectively CPO systems.

Chapter 0 introduces the supported web services and the corresponding service operations in detail. Chapter 3.5.3 defines the messages that will be exchanged between Hubject and partner systems. Chapter 4 specifies the data types of the messages. Chapter 5 details the required Data Types. Furthermore, every service is described in detail by a business process diagram, which is part of the appendix.



Introduction



1.6 Release management

With HBS 2.0, Hubject intends to pursue a new release rhythm (frequency) whereby multiple updates of the OICP could be done within a single year instead of the twice a year (i.e. guaranteed once a year on October 1st of each year) release rhythm pursed to date. The higher frequency of releases to be expected in the future should however not result in additional implementation complexity for partners since each new release will lead to a new version of the affected services that will run in parallel to the current service version. Consequently, partners do not need to perform an upgrade with each new release but will rather have the flexibility to choose which of the available valid service versions they wish to use. Service validity is dependent on the OICP version with which they are released (i.e. service versions available in deprecated OICP versions only are considered invalid and will not be supported by the HBS). Only service versions available in active OICP versions shall be supported. Each OICP version MUST be supported for two years by the HBS.

See below for a sample exhibit of the implications of the HBS 2.0 release management approach (*note: the dates and content in the diagram below are strictly meant for explanation purposes only. This is not to be considered as an official communication of release dates*):

				Current date: 15th Jan. 2019	
Available in:	OICP 2.0	OICP 2.1	OICP 2.2	OICP 2.3	OICP 2.4
Release Date OICP Version:	*Oct. 2015*	*Oct. 2016*	*Oct. 2017*	*Mar. 2018*	*Sep. 2018*
eRoamingService A	V1.0	V1.1	V2.0	V2.1	V2.2
eRoamingService B	V1.1	V2.0	V2.0	V2.0	V2.1
eRoamingService C	V1.0	V1.1	V1.2	V1.2	V1.2

- Service versions: in this sample exhibit, assuming the current date is 15th January 2019, only service versions highlighted in green will be supported by the HBS whereas services highlighted in grey will no longer be supported since these service versions are **only** available in deprecated OICP versions (i.e. the OICP versions highlighted in red). Consequently, partners can choose one of 3 valid service versions to implement for “Service A”, and one of 2 valid versions for “Service B”. “Service C” on the other hand has only one valid version which can be implemented.

- OICP versions: the OICP versions highlighted in red will not be supported by the HBS assuming the current date is 15th January 2019 since they are older than two years and therefore deprecated. Also, as depicted in the above exhibit and in contrast to the pre-HBS 2.0 release management approach, more than two OICP versions can be supported simultaneously going forward.

All partners using the HBS MUST indicate the OICP version they are currently running in their system (i.e. OICP version being used in the communication with the HBS). On the database level however, the current specifications are to be implemented in all systems, i.e. new mandatory fields should be filled with a value.

1.7 Further documents

To enable a fast and efficient connection process between the Hubject Brokering System and Partner Systems, the following documents contain further information.

- **support.hubject.com –** Contains all relevant information regarding the onboarding process and other technical information
- **WSDL files and REST API Documentation** (https://www.hubject.com/downloads/oicp/) – Technical API specifications
- **Dynamic Pricing - Functional Guide for Service Implementation -** contains all relevant details to enable a holistic view (i.e. technical and business perspectives) of how to best capitalize on the capabilities offered by the eRoamingDynamicPricing service.

1.8 OICP protocol version and service versions

Beginning with OICP 2.1, service endpoints are versioned individually and independent of the OICP version. The table below therefore gives an overview of all web services and their current version within OICP version 2.2.

Service	Version
eRoamingAuthorization	2.1
eRoamingReservation	1.1
eRoamingAuthenticationData	2.1
eRoamingEvseData	2.2
eRoamingEvseStatus	2.1
eRoamingDynamicPricing	1.0

1.9 Overview of Document Reviews

The table below provides an overview of all changes made to this document after its initial publication. The changes documented below affect descriptions provided in this document and are mostly corrections or refinements of the specification details.

Date of Update	Chapter Updated	Comments
15th June 2018	Chapter 5.1.31 – OpeningTimesType	• ***Nature of update*:** Informational • ***Details*:** specification refinement of the "*On*" attribute in the opening times data type
15th June 2018	Chapter 5.3.12 – LanguageCodeType	• ***Nature of update*:** Implementation relevant • ***Details*:** Update of the Regular Expression
15th June 2018	Chapter 5.1.2 – IdentificationType	• ***Nature of update*:** Informational • ***Details*:** specification refinement with additional details
15th June 2018	Chapter 5.2.16 – AdditionalReferenceType	• ***Nature of update*:** Informational • ***Details*:** specification refinement with additional details
15th June 2018	Chapter 4.1.2 - eRoamingPullEVSEData	• ***Nature of update:*** *Informational* • ***Details:*** specification refinement with additional details

2 Communication paradigms

2.1 SOAP and REST

The service communication between provider systems and Hubject is based on web-service communication using either SOAP 1.1 or REST. For each eRoaming service, the technical interface description is represented by WSDL files for the SOAP API and a pdf documentation for the REST API. The WSDL files and REST API pdf documentation supplement this OICP release document.

All services predating OICP 2.2 are available over both the SOAP and REST APIs. The eRoamingDynamicPricing service and all new services to be introduced in the future beginning with OICP 2.2 will be available over the REST API only. This means SOAP communication cannot be used for all new services introduced with OICP version 2.2 or later. Partners that wish to use these new services will therefore need to implement RESTful APIs for the respective communication with the HBS.

All web services described below are synchronous. All service messages exchanged between Hubject and partner systems MUST use UTF-8 character encoding.

2.2 Security

The Hubject system has been designed to be secure. As a consequence the following patterns are not allowed in the data of SOAP requests:

&gt	&lt	;	<	>
create	delete	drop	execute	insert
select	truncate	update		

Elements of the ChargingFacilityType are allowed to contain the characters ‘<’ and ‘>’ as an exception to this rule. The usage of spaces outside of quotes in a field is not allowed, if not stated otherwise.

2.3 Availability

The Hubject system will be set up in a highly available environment. Please check the partner contract for details.

2.4 Error handling

Service requests that are sent to Hubject over the SOAP API will be validated against the corresponding WSDL (technical service definition). In case that a request does not match the WSDL, Hubject will respond with a standard SOAP fault message indicating the format violation.

Communication paradigms

In case that a partner system cannot be addressed by Hubject, Hubject will monitor the connection error in the service session logging.

In case that a partner system does not respond to a request by Hubject within the internally defined period, Hubject will monitor the connection timeout in the service session logging.

General Hubject system errors that MAY occur during service processing will be caught. The system will then respond to the service requestor with a default eRoamingAcknowledgement message.

2.5 Status codes

Most service response messages contain a “StatusCode” field (e.g. eRoamingAcknowledgement (see 4.2.6)). The node provides a standardized code and status description that can be used to return details about certain process statuses. If for example an eRoamingAuthorizeStart request fails, the requested provider can e.g. specify why the user cannot be authorized.

Chapter 5.2.10 contains an overview of all relevant status codes.

The different states are standardized in order to make automated status processing possible. Backend systems only have to analyze the provided status code, irrespective of the functional status description.

The StatusCode node additionally contains the optional “AdditionalInfo” field. This field can be used to provide individual information or process details that go beyond the standardized description. In case that the optional “Description” field is used, the field should contain only defined values (see below).

2.6 Session handling

Some web service operations that are defined in chapter 0 together form a functional business process, respectively a functional session.

Example:
The operations in eRoamingAuthorization (see 0) cover a charging session. A charging session can be started with eRoamingAuthorizeStart or eRoamingAuthorizeRemoteStart operations and stopped with the corresponding operations. For this charging session, energy consumption data can then be sent using the eRoamingChargeDetailRecord operation.

To be able to relate individual operations to the correct session, Hubject assigns a SessionID to every session after the receipt of an initial request (e.g. eRoamingAuthorizeStart). The SessionID is part of the operation response and MUST be provided with each subsequent request that belongs to the session. In case that a request contains a SessionID that was not created by Hubject or one that is not valid, the request receives a negative response and no further process steps are performed.

Communication paradigms

Hubject uses globally unique identifiers (GUID) for SessionID creation. Furthermore, it is possible that partner backend systems use their own session concept. Hubject supports this by offering two (optional) request parameters CPOPartnerSessionID and EMPPartnerSessionID. CPO partner systems can use the CPOPartnerSessionID parameter to send their own session IDs. Hubject will assign the CPOPartnerSessionID to the Hubject SessionID and will add the CPOPartnerSessionID to every operation response so that the CPO partner systems can relate the operations to their own session handling.

The replacement of the “PartnerSessionID” parameter in OICP 2.1 with the two parameters “CPOPartnerSessionID” and “EMPPartnerSessionID” in OICP 2.2 has the following mapping implications for communication between OICP 2.1 and OICP 2.2 where applicable; “CPOPartnerSessionID” and “EMPPartnerSessionID” will be mapped to the “PartnerSessionID” and vice versa as and when necessary. Since “PartnerSessionID” in OICP 2.1 is defined as a string of 50 characters whereas “CPOPartnerSessionID” and “EMPPartnerSessionID” are strings of 250 characters, strings exceeding 50 characters will be cut where necessary during mapping.

Note:
Regarding eRoaming services, the SessionID will be the leading process identifier.

2.7 ProviderIDs and OperatorIDs

Most web service operations require the provision of a ProviderID (EMP) or OperatorID (CPO), depending on whether the operation is requested by EMPs or CPOs. The ProviderID is a composition of a country code and a three-digit string (see 5.3.3). The OperatorID is a composition of a country code and a three or three- to six-digit string (see 5.3.4), depending on whether the ID relates to the DIN or the ISO standard.

The country codes have been appended to the IDs in order to guarantee doubtless cross-national partner identification.

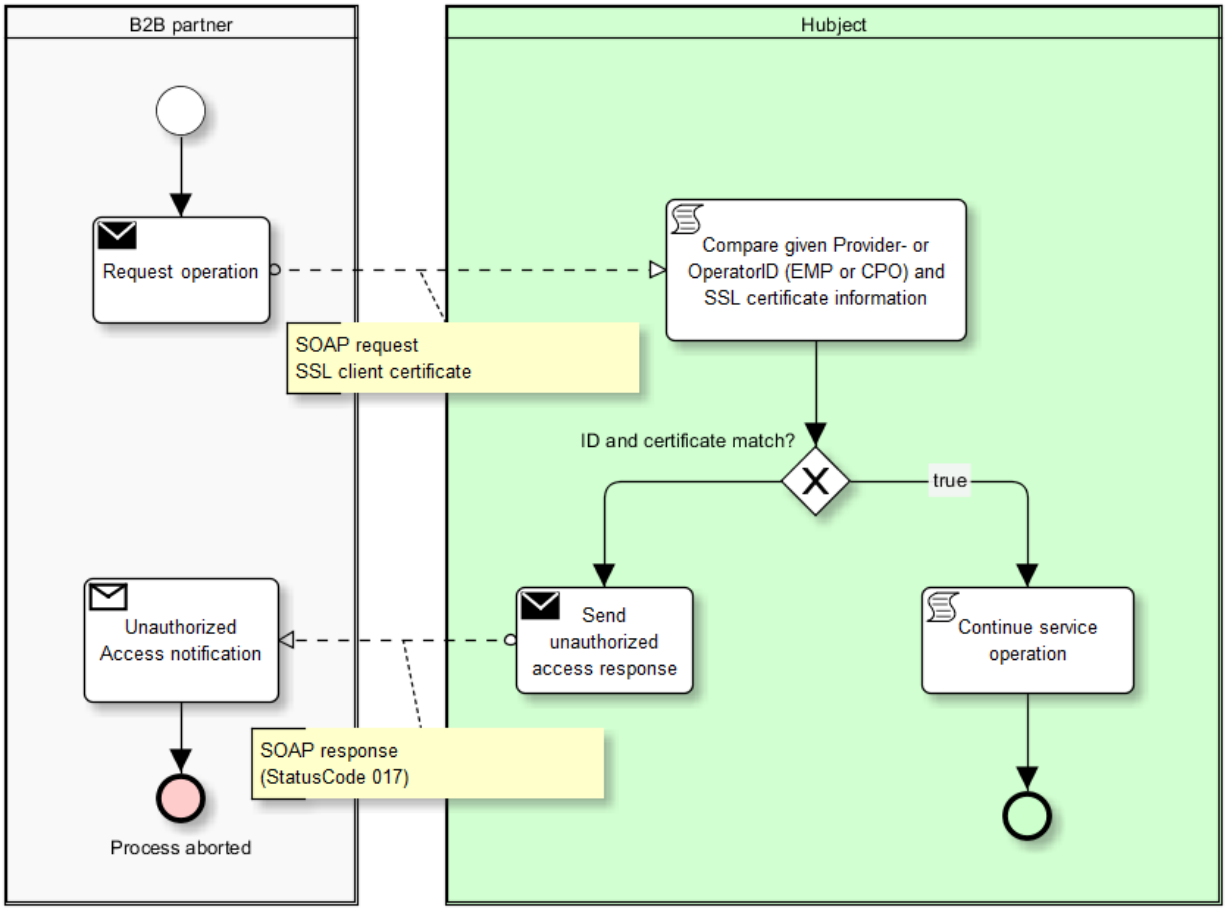
The IDs are assigned to a specific partner role and they are cross-market unique. There are two roles: eMobility provider (EMP) and charge point operator (CPO). A partner can have one or both roles. In case that a partner has both roles, two IDs (ProviderID see 5.3.3 and OperatorID see 5.3.4) will be assigned to the partner. Depending on which ID is provided with a service request, Hubject can identify the role that the partner has regarding the current service session.

If the appropriate ProviderID and OperatorID can be provided implicitly through Evco- or EvseIDs, EvcoIDs contain the corresponding EMP’s ProviderID whereas EvseIDs contain the corresponding CPO’s OperatorID.

With every web service request, Hubject compares the given Provider- or OperatorID to the partner’s SSL client certificate information. This way, Hubject ensures that a partner cannot request operations in the name of another partner by simply sending another partner’s ID. In case Hubject detects a mismatch between ProviderID/OperatorID and the client certificate information sent with the request, Hubject will not perform the operation and will respond with the status code 017 “Unauthorized Access”.

Communication paradigms

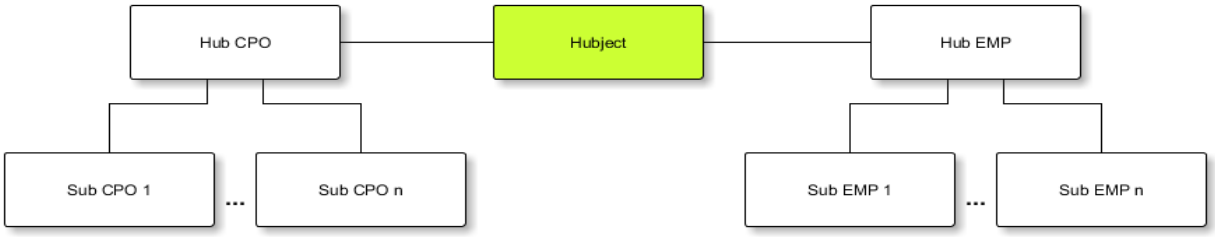
The following process diagram describes the partner identification process. It is performed at the beginning of every web service operation, which is described in this document. Consequently, all process diagrams in the appendix are implicitly preceded by this diagram.



Communication paradigms

2.8 Hub Provider and Hub Operator

Partners that are registered with Hubject have the possibility to bundle sub providers (EMP) or sub operators (CPO) and to act as “hub provider” or “hub operator”. Thus, the sub partners need not register with Hubject, because they will use the hub partner system in communicating with Hubject. The following diagram shows the relationships between Hubject, hub partners, and sub partners.



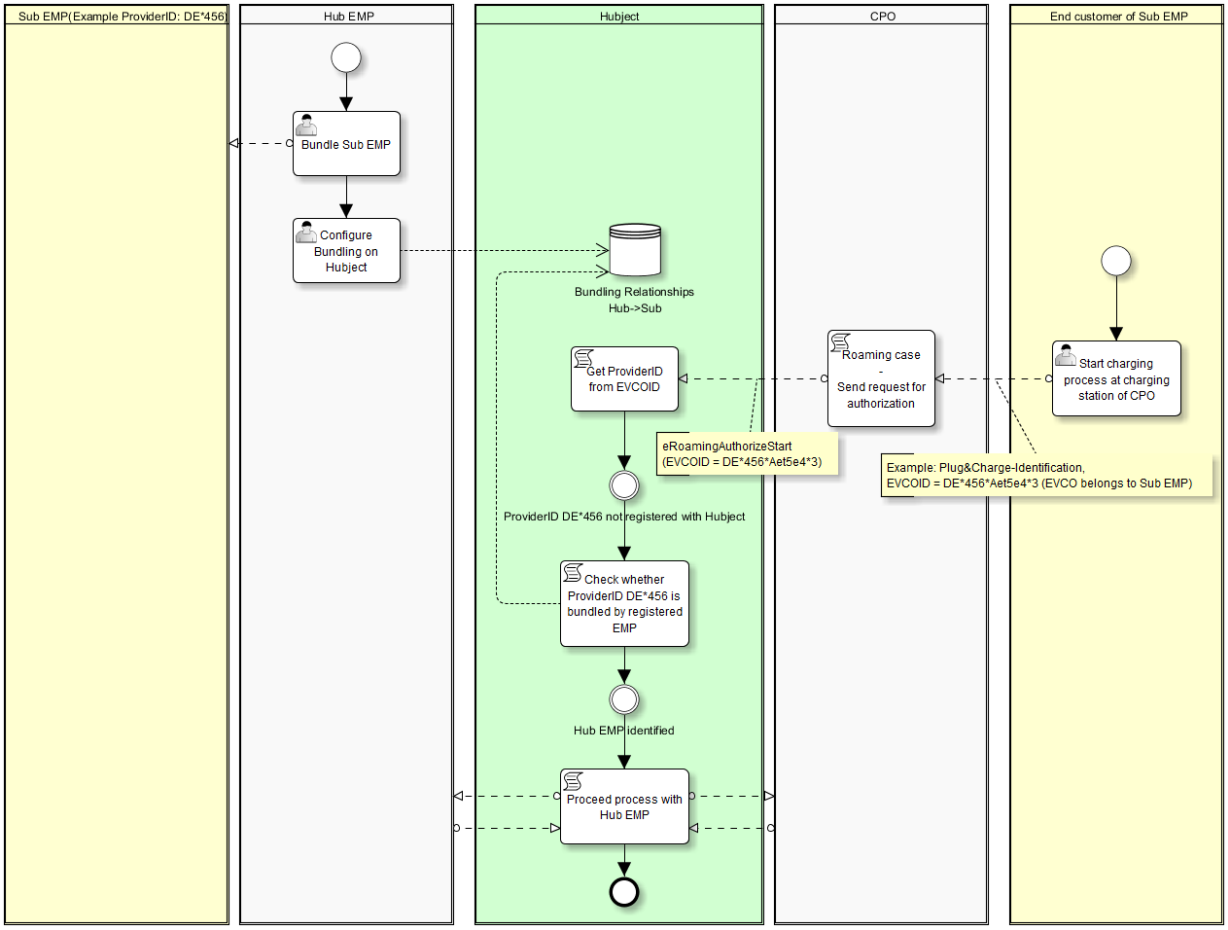
Hubject may receive service requests that contain sub partner information, e.g. an EvcoID containing the ProviderID of a sub partner. In such cases - when Hubject does not find the ID within the group of registered partners – Hubject will check whether the corresponding partner is bundled by a registered hub provider or hub operator. If so, the following service process will be conducted on behalf of the hub partner.

Note:
The web service fields ProviderID and OperatorID that are described in chapter 0 MUST always provide the ID of the partner communicating directly with Hubject. So, in case of a hub/sub scenario the fields always provide the ID of the hub partner. Sub partner IDs will only be provided implicitly through Evco- or EvseIDs.

Example:
The following diagram shows a sample scenario. A hub EMP bundles a sub EMP with the ProviderID “DE*456”. A customer of the sub EMP wants to charge a vehicle at a CPO’s charging station. The customer identifies himself via an EvcoID that contains the sub EMP’s ProviderID “DE*456”. Hubject cannot identify “DE*456” within the pool of EMPs that are registered with Hubject. Consequently, Hubject checks whether “DE*456” is bundled by a registered EMP. Hubject identifies the hub EMP and continues the process on behalf of the hub EMP. This means that e.g. an online authorization request or the forwarding of a charge detail record request will be sent to the hub EMP.

In case you are operating charging stations in different countries, please make sure each EVSE is equipped with the correct country code and the corresponding Operator ID. This Operator ID has to be either a sub-operator ID or your main ID.

Communication paradigms



2.9 Data push operations

Hubject offers different operations that allow partners to upload (push) data, e.g. upload of authentication data by EMPs or EVSE data by CPOs.

To guarantee data consistency, data push requests that address the same operation MUST always be processed sequentially. They MUST never be executed in parallel. This means that a partner system MUST always wait for the Hubject system's operation response before initiating the next request.

The reason for this is that push requests that are sent in parallel are also processed in parallel by Hubject. Thus, different requests may overtake each other and change their sequence before Hubject stores the data. This could lead to unintended data conditions.

Example:
A CPO sends an EVSE full load with several hundred EVSE records. Shortly after that, the same CPO sends an EVSE full load with only one EVSE record. From the CPO's point of view, the second request should overwrite the first, resulting in only one valid EVSE record. But probably the second (small) request will overtake the first (big) request. This results in the big request overwriting the small one and with it several hundred valid EVSE records on the Hubject system.

2.10 Time zones

Message fields that are specified by the field type “Date/Time” (e.g. “SessionStart” in eRoamingChargeDetailRecord, see 4.2.1) are technically assigned to the XML data type “datetime”. This type allows specifying a time zone by specifying an offset.

The time needs to be delivered in the format “complete date plus hours, minutes and seconds” referring to ISO 8601:1988 (E), with a time zone offset in hours and minutes. A time zone offset of "+hh:mm" indicates that the date/time uses a local time zone which is "hh" hours and "mm" minutes ahead of UTC. A time zone offset of "-hh:mm" indicates that the date/time uses a local time zone which is "hh" hours and "mm" minutes behind UTC.

YYYY-MM-DDThh:mm:ssTZD, e.g. “2014-02-01T15:45:00+02:00”,

where:

YYYY = four-digit year
MM = two-digit month (01=January, etc.)
DD = two-digit day of month (01 through 31)
T = separator
hh = two digits of hour (00 through 23) (am/pm NOT allowed)
mm = two digits of minute (00 through 59)
ss = two digits of second (00 through 59)
TZD = time zone designator (+hh:mm or -hh:mm)

Messages that are sent to Hubject and that Hubject directly passes to another partner (e.g. eRoamingChargeDetailRecord from CPO to EMP) will not be changed by Hubject (including time zone specifics).

Because of the need for time based charging fees it is mandatory for CPO’s to provide date time values including a time zone offset which refers to the charge point location.

Beginning with OICP 2.2, the HBS will store all Date/Time values in their original form. Also, the original Date/Time values as received and stored by the HBS will be provided in the response to requests from partner systems.

3 Services and Operations

Each message requires a message in return (at least an acknowledgement).

Each system must be able to cope with possible connection error scenarios as well as with different strategies to solve inconsistencies.

All described services are offered by Hubject. Some of the described services MUST also be offered by the connected provider systems because Hubject forwards incoming requests to partners which presumes that the corresponding service is offered by that partner (e.g. eRoamingAuthorization).

Other services will only be consumed by partner systems, meaning that these services do not have to be offered by the partner system (e.g. the eRoamingAuthenticationData service will be offered by Hubject only).

The names of all following services contain a version and are based on the pattern “<service>_V<version>”, e.g. “eRoamingAuthorization_V2.1”. The reason for this being that Hubject potentially has to offer different versions of a service at the same time. This requires different technical service endpoints (service URLs) and the version in the service name is used to enable the service endpoint differentiation.

3.1 eRoamingAuthorization_V2.1

The service eRoamingAuthorization contains several different operations. It MUST be offered by Hubject and the partner systems.

3.1.1 eRoamingAuthorizeStart

- Request message: eRoamingAuthorizeStart
- Response message: eRoamingAuthorizationStart

<u>Functional description:</u>

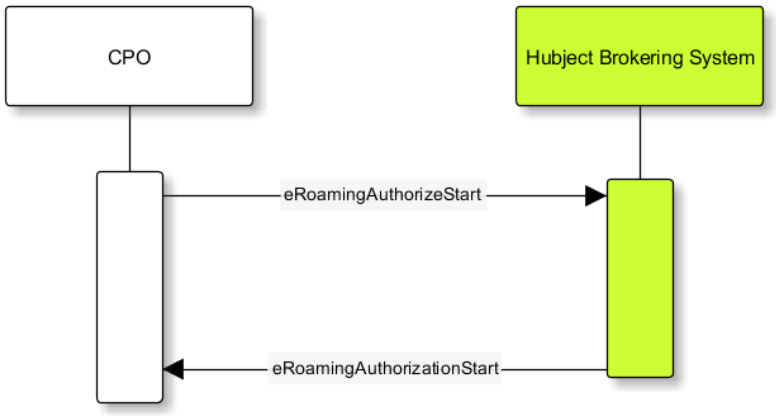
Scenario:
A customer of an EMP wants to charge a vehicle at a charging point of a CPO. The customer authenticates at the charging point. The CPO’s operator system does not recognize the customer’s authentication data. In order to authorize the charging process, the CPO’s system can send an eRoamingAuthorizeStart request to Hubject. The request MUST contain the OperatorID and the identification data (e.g. UID or EvcoID) and MAY contain the EvseID.

Hubject generates a SessionID for the charging process and persists important session data (SessionID, EvseID, identification data).

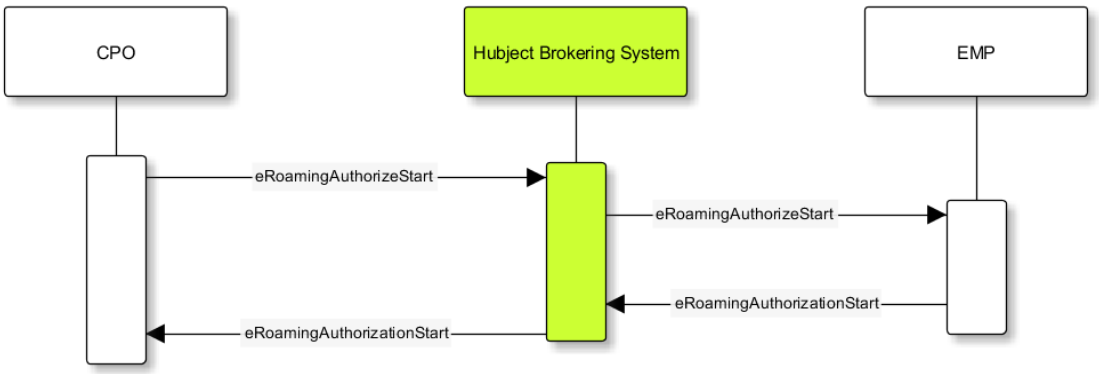
Regarding the further service processing, there are three different options:

A) Hubject first tries to authorize the customer offline by checking authentication master data. Authentication data can be uploaded by EMPs using the eRoamingAuthenticationData service. (see 3.3.1).

Services and Operations

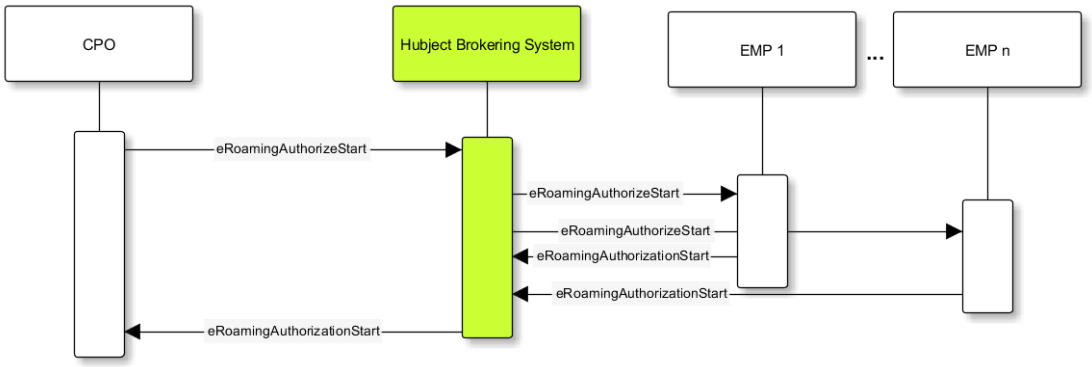


B) In case offline authorization is not possible, Hubject tries to derive the EMP from the provided identification data. QR Code and Plug&Charge identification data contain the EvcoID. Hubject can derive the EMP's ProviderID from the EvcoID. Hubject will directly forward eRoamingAuthorizeStart requests to the EMP. The EMP provider system checks the requested authentication data and responds accordingly, either by authorizing or not authorizing the request. The response MUST contain the ProviderID and the AuthorizationStatus and MAY contain a list of identification data that is authorized to stop the charging process. In case that the EMP provider system cannot be addressed (e.g. due to technical problems), the corresponding provider will be dealt with as if responding "NotAuthorized".



C) In case that Hubject cannot derive the EMP from the identification data (e.g. with RFID identification), Hubject identifies all EMPs that are under contract with the CPO (EMPs must be the service subscriber) and forwards the eRoamingAuthorizeStart request to all these EMPs (broadcast). Hubject consolidates all EMP responses and creates an overall response, authorizing the request in case that one EMP authorized the request.

Services and Operations



In case that the request for authorization was not successful, Hubject deletes the corresponding SessionID for the charging process.

The response from Hubject to the CPO contains authorization details and in case of successful authorization the created SessionID and the ProviderID of the authorizing provider.

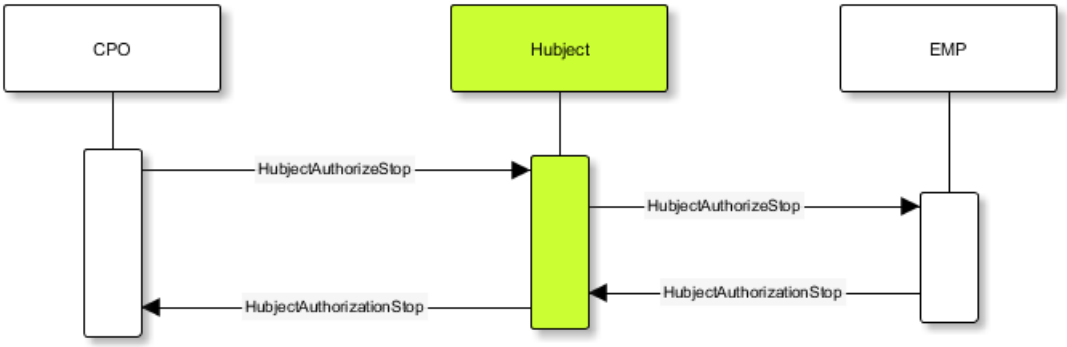
<u>Pin Security:</u>
The eRoamingAuthorizeStart request contains one of the defined identification types (see 5.1.2). The identification type “QRCodeIdentificationType” (see 5.1.11) contains - besides the “EvcoID” field - a “PIN” field or a “HashedPIN” field (only one of the two options must be provided).

For security reasons and as a general rule, Hubject does not store PINs in clear text, but always as encrypted hash values. In order to prevent hashed PIN values that may have been picked illegally from being used to request the authorization for charging processes, the PIN value MUST always be provided in clear text within the eRoamingAuthorizeStart request. This means that this operation MUST always provide the “PIN” field (clear text). Hubject will always generate a hash value of the provided PIN before checking the offline authentication data. So, in case that a PIN is provided by mistake as hashed value, Hubject automatically generates a hash of a hash, which eventually leads to a denial of authorization.

In order to create hash values, Hubject applies the hash algorithm that the EMP has assigned to the QR Code identification record (see 3.3.1).

3.1.2 eRoamingAuthorizeStop

- Request message: eRoamingAuthorizeStop
- Response message: eRoamingAuthorizationStop



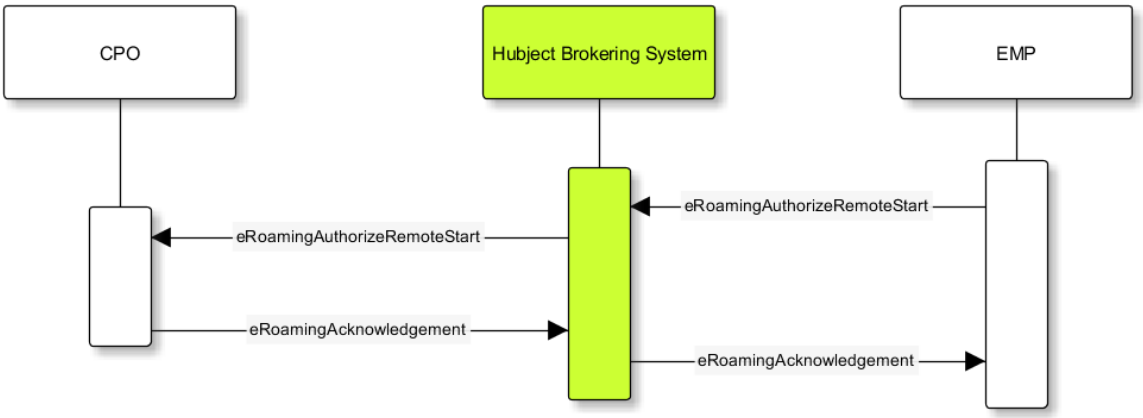
eRoamingAuthorizeStop basically works in a similar way to the operation eRoamingAuthorizeStart. The request is sent in order to authorize the stopping of a charging process. The request MUST contain the SessionID that was created by Hubject after the initial eRoamingAuthorizeStart request. In most cases, Hubject can derive the EMP that authorized the charging process from the SessionID and can directly and offline authorize the request or forward the request for stopping to the EMP. In case the charging session was originally authorized offline by the HBS, the session MUST only be stopped with the same medium, which was used for starting the session.

3.1.3 eRoamingAuthorizeRemoteStart

Note:
This operation is used by EMPs in order to remote start a charging process.

The service that is offered by Hubject in order to allow customers to directly start a charging process via mobile app.

- Request message: eRoamingAuthorizeRemoteStart
- Response message: eRoamingAcknowledgement



Services and Operations

Functional description:

Scenario:
A customer of an EMP wants to charge a vehicle at a charging station of a CPO. The customer informs his EMP of his intention, e.g. via mobile phone or smart phone application. The EMP's provider system can then initiate a charging process at the CPO's charging station by sending an eRoamingAuthorizeRemoteStart request to Hubject. The request MUST contain the ProviderID and the EvseID.

Hubject will derive the CPO's OperatorID from the EvseID.

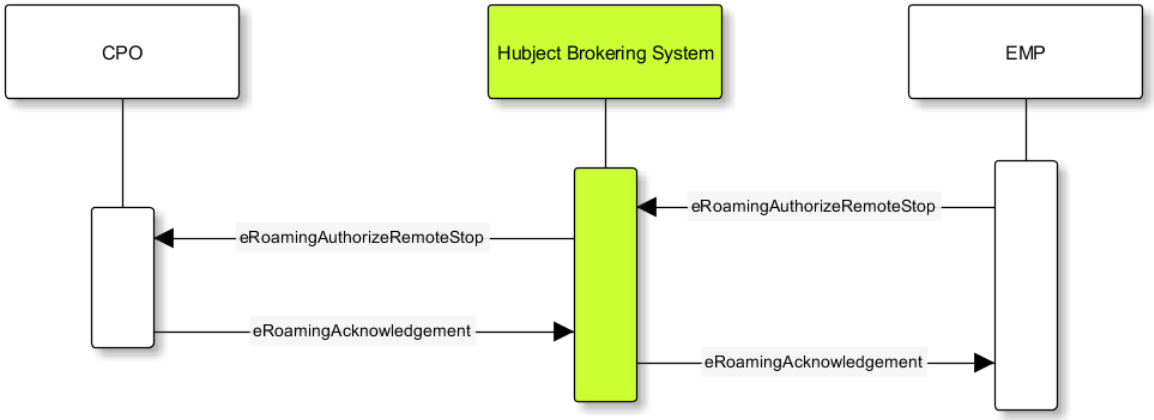
Hubject will check whether there is a valid contract between the two partners for the service (EMP must be the subscriber). If so, Hubject continues with checking the charging point compatibility. In case that the CPO has uploaded at least one charging point data record, Hubject will check whether the requested EvseID is among the uploaded data. If not, Hubject will respond with the status code 603 "Unknown EvseID". If yes, Hubject will check whether the charging spot's property "IsHubjectCompatible" is set "true". If the property is false, Hubject will respond with the status code 604 "EvseID is not Hubject compatible".

In case that the requested EvseID is compatible or the CPO has not uploaded any EVSE records at all, Hubject generates a SessionID for the following process and forwards the request (including the SessionID) to the CPO. The CPO MUST return an eRoamingAcknowledgement message that MUST contain the result indicating whether the charging process will be started and that MAY contain a status code for further information.

In case that the CPO's system cannot be addressed (e.g. due to technical problems), Hubject will return to the requestor a "false" result and a message indicating the connection error.

3.1.4 eRoamingAuthorizeRemoteStop

- Request message: eRoamingAuthorizeRemoteStop
- Response message: eRoamingAcknowledgement

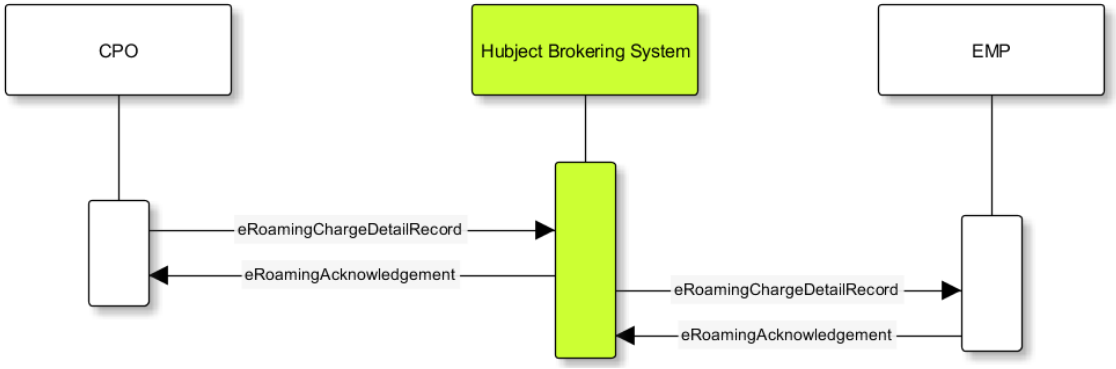


eRoamingAuthorizeRemoteStop basically works in the same way as eRoamingAuthorizeRemoteStart.

The only difference is that this request is sent in order to initiate the stopping of a charging process. The request MUST contain the SessionID that was created by Hubject after the initial eRoamingAuthorizeRemoteStart request.

3.1.5 eRoamingChargeDetailRecord

- Request message: eRoamingChargeDetailRecord
- Response message: eRoamingAcknowledgement



Functional description:

Scenario:
A customer of an EMP has charged a vehicle at a charging station of a CPO. The charging process was started with an eRoamingAuthorizeStart or an eRoamingAuthorizeRemoteStart operation. The process may have been stopped with an eRoamingAuthorizeStop or an eRoamingAuthorizeRemoteStop operation. A preceding stop request is not a necessary precondition for the processing of an eRoamingChargeDetailRecord request. The CPO's provider system MUST send an eRoamingChargeDetailRecord (CDR) after the end of the charging process in order to inform the EMP of the charging session data (e.g. meter values and consumed energy) and further charging process details.

Note:
The CPO MUST provide the same SessionID that was assigned to the corresponding charging process. Based on this information Hubject will be able to assign the session data to the correct process.

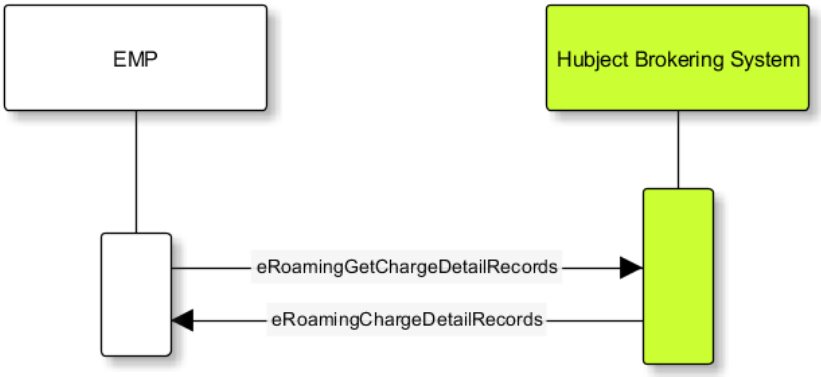
Hubject will identify the receiving EMP and will forward the CDR to the corresponding EMP. The EMP MUST return an eRoamingAcknowledgement message that MUST contain the result indicating whether the session data was received successfully and that MAY contain a status code for further information.

Hubject will accept only one CDR per SessionID.

In addition to forwarding the CDR to the EMP, Hubject also stores the CDR. In case that the recipient provider's system cannot be addressed (e.g. due to technical problems), Hubject will nevertheless return to the requestor a positive result provided that storing the CDR was successful.

3.1.6 eRoamingGetChargeDetailRecords

- Request message: eRoamingGetChargeDetailRecords
- Response message: eRoamingChargeDetailRecords



The operation allows EMPs to download CDRs that have been sent to Hubject by partner CPOs. This means if for example Hubject was unable to forward a CDR from a CPO to an EMP due to technical problems in the EMP's backend (see 3.1.5), the EMP will still have the option of obtaining these CDRs. The EMP MUST specify a date range in the request. Hubject will return a list of all CDRs received by the HBS within the specified date range for the requesting EMP (i.e. all CDRs within the date range where the corresponding charging process was authorized by the EMP or authorized by Hubject based on the EMP's authentication data).

Hubject does not check whether a requested CDR has already been provided to the requesting EMP in the past.

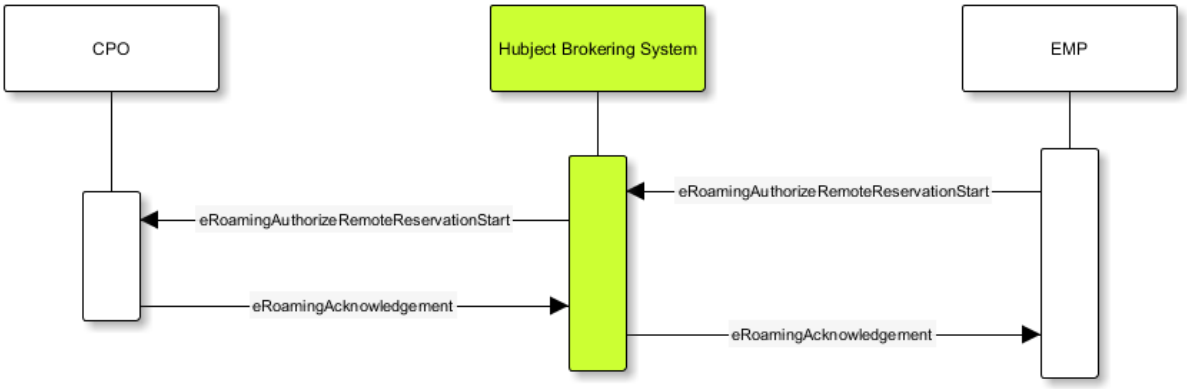
3.2 eRoamingReservation_V1.1

The service eRoamingReservation contains two operations. It MUST be offered by Hubject and MAY be offered by CPO partner systems. The Service MUST be enabled by Hubject for the CPO. If the charging station offers reservation services, the CPO can provide this information in the field ValueAddedServices (see 5.1.17).

3.2.1 eRoamingAuthorizeRemoteReservationStart

Note:
This operation is used by EMPs in order to remotely reserve a charging point.

- Request message: eRoamingAuthorizeRemoteReservationStart
- Response message: eRoamingAcknowledgement



Functional description:

Scenario:
A customer of an EMP wants to reserve a charging point of a CPO for a later charging process. The customer informs his EMP of his intention, e.g. via mobile phone or smart phone application. The EMP's provider system can then initiate a reservation of the CPO's charging point by sending an eRoamingAuthorizeRemoteReservationStart request to Hubject. The request MUST contain the ProviderID and the EvseID. The demanded reservation product can be specified using the field PartnerProductID.

Hubject will derive the CPO's OperatorID from the EvseID.

Hubject will check whether there is a valid contract between the two partners for the service Reservation (EMP must be the subscriber). If so, Hubject continues with checking the charging point compatibility. In case that the CPO has uploaded at least one charging point data record, Hubject will check whether the requested EvseID is among the uploaded data. If not, Hubject will respond with the status code 603 "Unknown EvseID". If yes, Hubject will check whether the charging spot's property "IsHubjectCompatible" is set "true". If the property is false, Hubject will respond with the status code 604 "EvseID is not Hubject compatible".

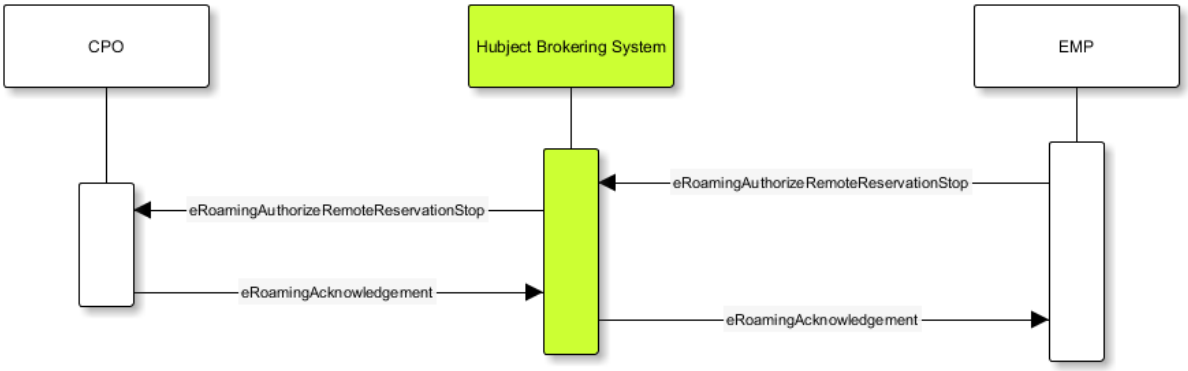
Services and Operations

In case that the requested EvseID is compatible or the CPO has not uploaded any EVSE records at all, Hubject generates a SessionID for the reservation process and forwards the request (including the SessionID) to the CPO. The CPO MUST return an eRoamingAcknowledgement message that MUST contain the result indicating whether the reservation was successful and that MAY contain a status code for further information.

In case that the CPO's system cannot be addressed (e.g. due to technical problems), Hubject will return to the requestor a "false" result and a message indicating the connection error.

3.2.2 eRoamingAuthorizeRemoteReservationStop

- Request message: eRoamingAuthorizeRemoteReservationStop
- Response message: eRoamingAcknowledgement



eRoamingAuthorizeRemoteReservationStop basically works in the same way as RoamingAuthorizeRemoteReservationStart.

The only difference is that this request is sent in order to end the reservation of a charging spot. The request MUST contain the SessionID that was created by Hubject after the initial eRoamingAuthorizeRemoteReservationStart request. After the eRoamingAuthorizeRemoteReservationStop the CPO MUST provide a CDR.

3.3 eRoamingAuthenticationData_V2.1

This service is only offered by Hubject.

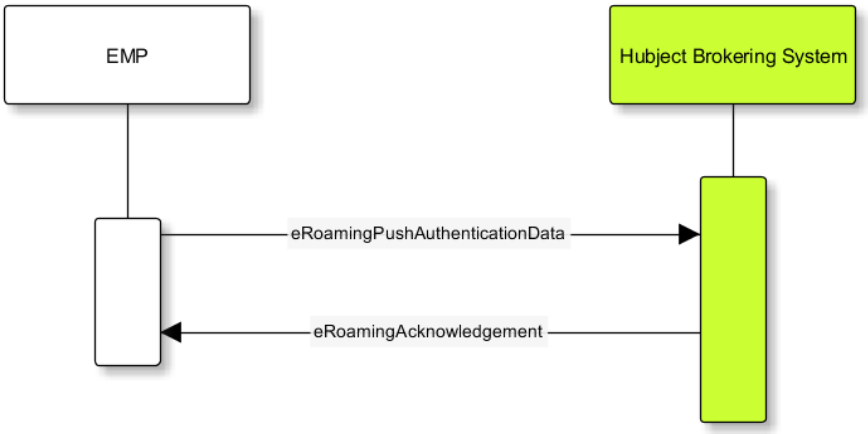
In addition to the online authorization service that requests customer authentication data on demand from the connected partner systems, Hubject offers the possibility to upload and download authentication data and thus to exchange data between different partners. EMPs can decide whether they disclose their authentication data for CPOs or not.

If an EMP uploads their data to Hubject, Hubject can authorize requests from other partners (e.g. CPOs) without having to forward the request to the EMP. The eRoamingPushAuthenticationData operation gives EMPs the possibility to upload (push) authentication data to the HBS.

Furthermore, Hub EMPs may also push authentication data of sub-EMPs. Hubject does not distinguish between authentication records of hub providers and their related sub providers (see 2.8).

3.3.1 eRoamingPushAuthenticationData

- Request message: eRoamingPushAuthenticationData
- Response message: eRoamingAcknowledgement



When an EMP sends an eRoamingPushAuthenticationData request, Hubject checks whether there is a valid contract between Hubject and the EMP for the service type (Hubject must be the subscriber). If so, the operation allows uploading authentication data to Hubject. Furthermore, it is possible to update authentication data that has been pushed with an earlier operation request. How Hubject handles the transferred data MUST be defined in the request field “ActionType”, which offers four options (see below). The authentication data to be inserted or updated MUST be provided with the “ProviderAuthenticationData” field, which consists of “AuthenticationDataRecord” structures (see 5.1.8 and 5.1.9). Hubject keeps a history of all updated and changed data records. Every successful push operation – irrespective of the performed action – leads to a new version of currently valid data records. Furthermore, each operation is logged with the current timestamp. Thus, Hubject can reconstruct the status of authentication data for every point in time in the past.

Services and Operations

Action types:

fullLoad:
The EMP uploads the full set of current authentication data. Hubject does not compare the new data to old (earlier pushed) data. It keeps a history of old data records and handles the newly provided data as valid. In order to allow an easy deletion of all records, it is possible to perform a fullLoad with an empty list of records.

insert
The EMP adds further authentication data records to the current set of data. Hubject verifies that the provided data records do not already exist in the currently valid data status. If so, the transaction will be aborted, no data will be inserted, and the request will be answered with an error message. Error details will be provided with the "AdditionalInfo" field (see 5.1.1).

update
The EMP updates data records of the current set of data. Hubject verifies that the provided data records do exist in the currently valid data status. If not, the transaction will be aborted, no data will be updated, and the request will be answered with an error message.

delete
The EMP deletes data records of the current set of data.

PIN security:
The authentication data records that are uploaded to Hubject contain one of the defined identification types (see 5.1.2). The identification type "QRCodeIdentificationType" (see 5.1.11) contains – besides an "EvcoID" field – a "PIN" field or a "HashedPIN" field (only one of the two options must be provided). For security reasons, Hubject generally does not store PINs in clear text, but always as encrypted hash values. When uploading authentication data to Hubject, the EMPs can directly provide hashed PIN values (using the field "HashedPIN"). In case that the PINs are provided in clear text (field "PIN"), Hubject will generate a hash value for every PIN and will store only the hashes. Hubject by default generates a hash using Bcrypt as a hashing function.

In case that an EMP provides already hashed PINs, he MUST also specify the corresponding hash generation algorithm so that Hubject can reproduce the hash generation when processing a request for authorization (see 3.1.1). For this reason, the "HashedPIN" field contains detailed information concerning the hash function and the hash salt value (for salted hash functions) that must be used for hash generation.

EVCO consistency:
EvcoIDs contain the ID of the corresponding EMP (see 5.3.1). With every data upload operation Hubject checks whether the given EMP's ProviderID (or Sub-ProviderIDs if necessary (see 2.8)) matches every given EvcoID. If not, Hubject refuses the data upload and responds with the status code 019.

Note:
The eRoamingPushAuthenticationData operation MUST always be used sequentially (see 2.9).

3.4 eRoamingEVSEData_V2.2

Hubject offers the possibility to upload and download charging spot (EVSE) data and, thus, to exchange data between different partners.

See appendix 6.6 for a detailed business process diagram regarding the EVSE data service.

The eRoamingPullEVSEData gives the EMPs the possibility to download (pull) EVSE data from partner operators via Hubject.

Hub CPOs (see 2.8) may also push EVSE data of sub operators. Hubject does not distinguish between EVSE records of hub operators and related sub operators.

3.4.1 eRoamingPullEVSEData

- Request message: eRoamingPullEVSEData
- Response message: eRoamingEVSEData

When an EMP sends an eRoamingPullEVSEData request, Hubject checks whether there is a valid contract between Hubject and the EMP for the service type (EMP must be the subscriber). If so, the operation allows downloading EVSEData from Hubject. When an EMP sends an eRoamingPullEVSEData request, Hubject identifies all currently valid EVSEData records of all operators.

Hubject groups all resulting EVSEData records according to the related CPO. The response structure contains an “EVSEData” node that envelopes an “OperatorEVSEData” node for every CPO with currently valid and accessible data records.

For every EVSE data record Hubject identifies the timestamp of the last update, which has been performed on the record. The timestamp is returned with the attribute “lastUpdate”.

Services and Operations

Delta pull

As mentioned above, the operation by default returns all currently valid EVSE data records. However, the requesting EMP has the possibility to download only the changes (delta) compared to a certain time in the past. In order to do so, the EMP MUST provide the optional date/time field “LastCall” (see 4.1.2), indicating his last EVSE pull request. In case that Hubject receives the LastCall parameter, Hubject compares the EVSE records from the time of the last call with the currently valid records. As a result, Hubject assigns the attribute “deltaType” (possible values: insert, update, delete) to every response EVSE data record (see 5.1.17) indicating whether the particular record has been inserted, updated or deleted in the meantime. EVSE data records that have not changed will not be part of the response.

Note:
The delta pull option cannot be combined with radial search, because in some cases this could lead to data inconsistency on the EMP’s side. This is why the API only allows the provision of either the attribute “SearchCenter” or “LastCall”.

3.5 eRoamingEVSEStatus_V2.1

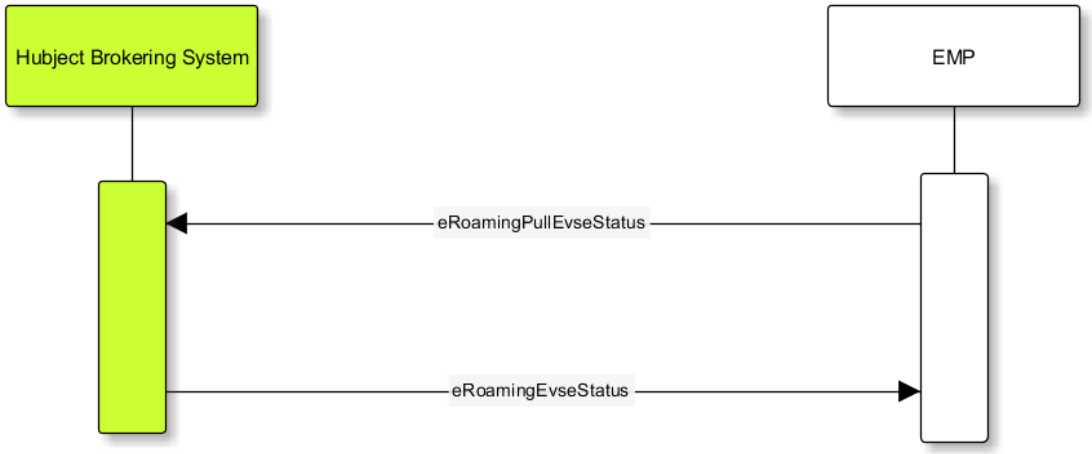
Hubject offers the possibility to upload and download dynamic charging spot (EVSE) status information and thus to exchange the data between different partners.

The eRoamingEVSEStatus service offers two operations: eRoamingPushEVSEStatus in order to give CPOs the possibility to upload (push) EVSEStatus data and eRoamingPullEVSEStatus in order to give EMPs the possibility to download (pull) EVSE status data from partner operators via Hubject.

Hub CPOs (see 2.8) may also push EVSE status records of sub operators. Hubject does not distinguish between EVSEStatus records of hub operators and related sub operators.

3.5.1 eRoamingPullEVSEStatus

- Request message: eRoamingPullEVSEStatus
- Response message: eRoamingEVSEStatus



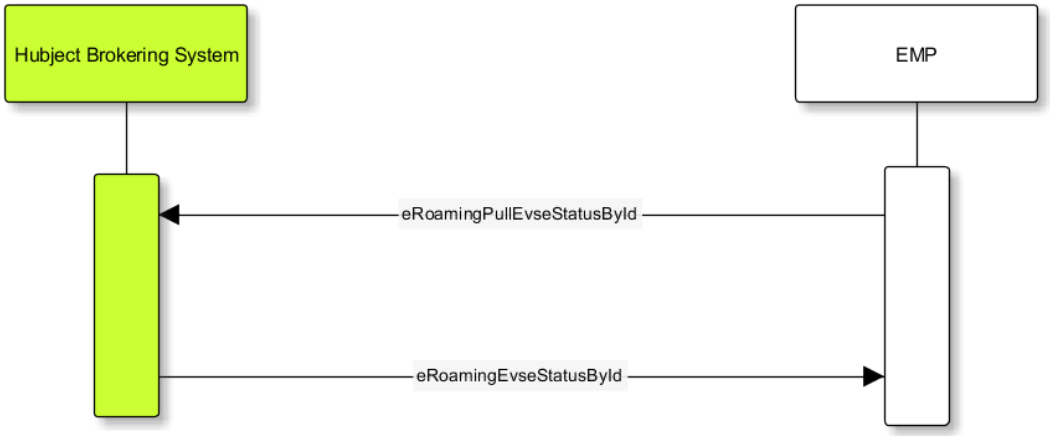
Services and Operations

When an EMP sends an eRoamingPullEVSEStatus request, Hubject checks whether there is a valid contract between Hubject and the EMP for the service type (EMP must be the subscriber). If so, the operation allows downloading EVSE status data from Hubject. When an EMP sends an eRoamingPullEVSEStatus request, Hubject identifies all currently valid EVSE status records of all operators.

Hubject groups all resulting EVSE status records according to the related CPO. The response structure contains an "EVSEStatuses" node that envelopes an "OperatorEVSEStatus" node for every CPO with currently valid and accessible status data records.

3.5.2 eRoamingPullEVSEStatusById

- Request message: eRoamingPullEVSEStatusById
- Response message: eRoamingEVSEStatusById



The operation works similar to the above described eRoamingPullEVSEStatus operation. However, in contrast to the eRoamingPullEVSEStatus operation, the eRoamingPullEVSEStatusById operation requires that a list of EvseIDs is sent with the request message. For every requested EvseID, Hubject identifies the currently valid EVSE status. The resulting EVSE records are not grouped according to the related CPO as it is done with the eRoamingPullEVSEStatus operation. In case that a requested EvseID does not exist in the Hubject database Hubject sets the value of the corresponding response field EVSEStatus to "EVSENotFound".

The requested list of EvseIDs MUST not contain more than 100 EvseIDs.

3.5.3 eRoamingPullEVSEStatusByOperatorID

- Request message: eRoamingPullEVSEStatusByOperatorID
- Response message: eRoamingEVSEStatusByOperatorID

Starting with OICP 2.2, the HBS offers EMPs the possibility to pull (download) EVSE status data for specific operators (CPOs). The operation works similar to the above described eRoamingPullEVSEStatusByID operation, except this operation requires that a list of OperatorIDs is sent with the request message. For every requested OperatorID, Hubject identifies the currently valid EVSE status data. The identified EVSE status records are grouped by OperatorID as it is done with the eRoamingPullEVSEStatus operation. In case a requested OperatorID does not

exist in the Hubject database, Hubject sets the value of the corresponding response field EVSEStatus to “EVSENotFound”.

3.6 eRoamingDynamicPricing_V1.0

The HBS with OICP 2.2 offers the possibility to flexibly or dynamically price Authorization services. The service mainly enables pushing (upload) and pulling (download) of pricing data to and from the HBS through webservice requests and/or downloads/uploads in the Hubject portal.

Flexible pricing enables CPOs to offer differentiated pricing (in multiple currencies) of charging processes at their charging stations. The differentiation of prices can be done along relevant dimensions such as charging facility characteristics (e.g. maximum charging power), EVSE location and time. OICP 2.2 offers the possibility to exchange this flexible price differentiation over webservice communication between CPOs and EMPs.

In addition to the above flexible pricing capability, OICP 2.2 enables more frequent and near real-time update and exchange of pricing information between CPOs and EMPs. This is achieved whereby a CPO continuously pushes valid prices to the HBS for specific EMPs to pull these prices.

With OICP 2.2, the HBS offers an extensive breadth of technical capabilities for flexible and dynamic pricing of charging and reservation services for CPOs. Please refer to the supplementary document “*Dynamic Pricing - Functional Guide for Service Implementation”* for more details and a holistic view (technical and business perspectives) of how to best capitalize on the capabilities offered by the eRoamingDynamicPricing service.

The eRoamingDynamicPricing service offers four operations, namely the:

1. eRoamingPushPricingProductData operation which gives CPOs the possibility to upload (push) pricing product information. Pricing products refer to the different tariffs offered by a CPO based on the differentiation dimensions mentioned above.

2. eRoamingPushEVSEPricing operation which gives CPOs the possibility to assign their various pricing products to individual EVSEs and thereby upload (push) location/EVSE-specific pricing data.

3. eRoamingPullPricingProductData operation which gives EMPs the possibility to download (pull) pricing product information uploaded by a CPO for the respective EMP.

4. eRoamingPullEVSEPricing operation which gives EMPs the possibility to download (pull) location/EVSE-specific pricing data uploaded by CPOs for the respective EMP.

3.6.1 eRoamingPullPricingProductData

- Request message: eRoamingPullPricingProductData
- Response message: eRoamingPricingProductData

When an EMP sends an eRoamingPullPricingProductData request, Hubject checks whether there is a valid flexible/dynamic pricing business contract (for the service type Authorization) between the EMP and the CPOs whose OperatorIDs are sent in the request. If so, the operation allows the download of pricing product data pushed to the HBS by these CPOs for the requesting EMP. When this request is received from an EMP, currently valid pricing products data available in the HBS for the requesting EMP (and pushed by CPOs whose OperatorIDs are supplied in the request) are grouped by OperatorID and sent in response to the request.

The operation also allows the use of the LastCall filter. When the LastCall filter is used, only pricing product data changes that have taken place after the date/time value provided in the “LastCall" field of the request are sent to the EMP.

3.6.2 eRoamingPullEVSEPricing

- Request message: eRoamingPullEVSEPricing
- Response message: eRoamingEVSEPricing

When an EMP sends an eRoamingPullPricingProductData request, Hubject checks whether there is a valid flexible/dynamic pricing business contract (for the service type Authorization) between the EMP and the CPOs whose OperatorIDs are sent in the request. If so, the operation allows the download of EVSE pricing data pushed to the HBS by these CPOs for the requesting EMP. When this request is received from an EMP, currently valid EVSE pricing data available in the HBS for the requesting EMP are grouped by OperatorID and sent in response to the request.

The operation also allows the use of the LastCall filter. When the LastCall filter is used, only EVSE pricing data changes that have taken place after the date/time value provided in the “LastCall" field of the request are sent to the EMP.

4 Messages

This chapter describes the messages and embedded information. The column M/O states, if the information is mandatory or optional.

4.1 Mandatory messages to be sent by EMPs

4.1.1 eRoamingPushAuthenticationData

eRoamingPushAuthenticationData is a message that is sent in order to upload authentication data to Hubject.

Please note:

This message is only for EMPs onboarded to the Hubject platform as offline EMPs.

EMP → HBS		**Related Service Version: V_2.1**	
Request: eRoamingPushAuthenticationData		**Response: eRoamingAcknowledgement (see 4.2.6)**	
Direction: EMP MUST send message to HBS		**Implementation: mandatory**	
Recommended frequency: daily			
Name	**Data Type**	**Description**	**M/O**
ActionType	One of: ▪ fullLoad ▪ update ▪ insert ▪ delete	Describes the action that has to be performed by Hubject with the provided data.	M
ProviderAuthenticationData	ProviderAuthenticationDataType (see 5.1.8)	Provider information	M

Messages

4.1.2 eRoamingPullEVSEData

eRoamingPullEVSEData is a message that is sent in order to request the download of EVSE data of operators stored on the Hubject system.

EMP → HBS	**Related Service Version: V_2.2**
Request: eRoamingPullEVSEData	**Response: eRoamingEVSEData (see 4.2.4)**
Direction: EMP MUST send message to HBS	**Implementation: mandatory**
Recommended frequency: daily	

Name	Data Type	Description	M/O
ProviderID	ProviderIDType (see 5.3.3)	Identifies the provider	M
SearchCenter	SearchCenterType (see 5.1.15)	The data can be restricted using search parameters that are provided in this field. Cannot be combined with “LastCall”.	O
LastCall	Date/Time	In case that this field is set, Hubject does not return the currently valid set of EVSE data but the changes compared to the status of EVSE data at the time of the last call. Cannot be combined with “SearchCenter”, “CountryCodes”, and “OperatorIDs”.	O
GeoCoordinatesResponseFormat	GeoCoordinates ResponseFormatType (see 5.2.7)	Defines the format of geo coordinates that shall be provided with the response.	M
CountryCodes	CountryCodeType (see 5.2.11)	A list of countries whose EVSE’s a provider wants to retrieve. Cannot be combined with “LastCall”.	O
OperatorIDs	OperatorIDType (see 5.3.4)	A list of Operator Ids in ISO or DIN standard to download only EVSE’s of one or more operators. Cannot be combined with “LastCall”.	O

4.1.3 eRoamingPullEVSEStatus

eRoamingPullEVSEStatus is a message that is sent in order to request the download of EVSE status data stored on the Hubject system.

EMP → HBS	**Related Service Version: V_2.1**
Request: eRoamingPullEVSEStatus	**Response: eRoamingEVSEStatus (see 4.2.5)**
Direction: EMP MUST send message to HBS	**Implementation: mandatory**

Name	Data Type	Description	M/O
ProviderID	ProviderIDType (see 5.3.3)	Identifies the provider	M
SearchCenter	SearchCenterType (see 5.1.15)	The data can be restricted using search parameters, which are provided in this field.	O
EVSEStatus	EVSEStatusType (see 5.2.9)	Status of the EVSE	O

In case not all but a specific EVSE status is needed, Hubject offers the service eRoamingPullEVSEStatusByID (see 4.3.1) and eRoamingPullEVSEStatusByOperatorID (see 4.3.4).

Messages

4.1.4 eRoamingAuthorizeRemoteStart

eRoamingAuthorizeRemoteStart is a message to request an authorization for starting a charging process.

EMP → HBS		**Related Service Version: V_2.1**		
Request: eRoamingAuthorizeRemoteStart		**Response: eRoamingAcknowledgement (see 4.2.6)**		
Direction: EMP MUST send message to HBS		**Implementation: mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process.	O	
CPOPartnerSessionID	String	Optional field containing the session ID assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
EMPPartnerSessionID	String	Optional field containing the session ID assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
ProviderID	ProviderIDType (see 5.3.3)	The ProviderID is defined by Hubject and is used to identify the EMP.	M	
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	M	
Identification	IdentificationType (see 5.1.2)	Authentication data.	M	
PartnerProductID	ProductIDType (see 5.2.12)	A pricing product name (for identifying a tariff) that must be unique	O	

Messages

4.1.5 eRoamingAuthorizeRemoteStop

eRoamingAuthorizeRemoteStop is a message to request an authorization for stopping a charging process.

EMP → HBS		**Related Service Version: V_2.1**		
Request: eRoamingAuthorizeRemoteStop		**Response: eRoamingAcknowledgement (see 4.2.6)**		
Direction: EMP MUST send message to HBS		**Implementation: mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process.	M	
CPOPartnerSessionID	String	Optional field containing the session ID assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
EMPPartnerSessionID	String	Optional field containing the session ID assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
ProviderID	ProviderIDType (see 5.3.3)	The ProviderID is defined by Hubject and is used to identify the operating provider.	M	
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	M	

Messages

4.1.6 eRoamingGetChargeDetailRecords

eRoamingGetChargeDetailRecords is a message to request a list of charge detail records.

Please note:
This message is only mandatory for offline EMPs.

EMP → HBS		**Related Service Version: V_2.1**	
Request: eRoamingGetChargeDetailRecords		**Response: eRoamingChargeDetailRecords (see 4.2.1)**	
Direction: EMP MUST send message to HBS		**Implementation: mandatory**	
Recommended frequency: daily			
Name	**Data Type**	**Description**	**M/O**
ProviderID	ProviderIDType (see 5.3.3)	The ProviderID is defined by Hubject and is used to identify the EMP.	M
From	Date/Time	Start of the requested time range.	M
To	Date/Time	End of the requested time range.	M

Messages

4.1.7 eRoamingAuthorizationStart

eRoamingAuthorizationStart is a message that authorizes a user to charge a car.

Please note:
This message describes the response which has to be sent in response to the eRoamingAuthorizeStart and is only mandatory for online EMPs.

EMP → HBS		**Related Service Version: V_2.1**		
Request: eRoamingAuthorizeStart (see 4.2.2)		**Response: eRoamingAuthorizationStart**		
Direction: EMP MUST send message to HBS		**Implementation: mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process (in case of successful authorization).	O	
CPOPartnerSessionID	String	Optional field containing the session ID assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
EMPPartnerSessionID	String	Optional field containing the session ID assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
ProviderID	ProviderIDType (see 5.3.3)	The ProviderID is defined by Hubject and is used to identify the EMP. In case of a positive authorization this field will be filled.	O	
AuthorizationStatus	AuthorizationStatusType (see 5.2.1)	Information specifying whether the user is authorized to charge or not.	M	
StatusCode	StatusCodeType (see 5.1.1)	Structured status details. Can be used to specify the reason for a failed authorization.	M	

Messages

EMP → HBS		Related Service Version: V_2.1		
Request: eRoamingAuthorizeStart (see 4.2.2)		**Response: eRoamingAuthorizationStart**		
Direction: EMP MUST send message to HBS		**Implementation: mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
AuthorizationStop Identifications	List (IdentificationType) (see 5.1.2)	A list of Identification data that is authorized to stop the charging process.	O	

4.1.8 eRoamingAuthorizationStop

eRoamingAuthorizeStop is a message to request an authorization for stopping a charging process.

Please note:
This message describes the response which has to be sent in return to the eRoamingAuthorizeStop request and is only mandatory for online EMPs.

EMP → HBS		Related Service Version: V_2.1		
Request: eRoamingAuthorizeStop (see 4.2.3)		**Response: eRoamingAuthorizationStop**		
Direction: EMP MUST send message to HBS		**Implementation: mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process.	O	
CPOPartnerSessionID	String	Optional field containing the session ID assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250

Messages

EMP → HBS		**Related Service Version: V_2.1**		
Request: eRoamingAuthorizeStop (see 4.2.3)		**Response: eRoamingAuthorizationStop**		
Direction: EMP MUST send message to HBS		**Implementation: mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
EMPPartnerSessionID	String	Optional field containing the session ID assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
ProviderID	ProviderIDType (see 5.3.3)	The ProviderID is defined by Hubject and is used to identify the EMP.	O	
AuthorizationStatus	AuthorizationStatus Type (see 5.2.1)	Information specifying whether the user is authorized to charge or not.	M	
StatusCode	StatusCodeType (see 5.1.1)	Structured status details. Can be used to specify the reason for a failed authorization.	M	

4.2 Mandatory messages to be received by EMPs

4.2.1 eRoamingChargeDetailRecord

eRoamingChargeDetailRecord is a message containing charging process details (such as meter values, etc.).

Please note:
This message is only mandatory for online EMPs.

HBS → EMP		**Related Service Version: V_2.1**		
Request: eRoamingChargeDetailRecord		**Response: eRoamingAcknowledgement (see 4.2.6)**		
Direction: **EMP MUST receive message from HBS**		**Implementation:** **mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process. Hubject will accept only one CDR per SessionID.	M	
CPOPartnerSessionID	String	Optional field containing the session ID assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
EMPPartnerSessionID	String	Optional field containing the session ID assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
PartnerProductID	ProductIDType (see 5.2.12)	A pricing product name (for identifying a tariff) that must be unique	O	
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	M	

Messages

HBS → EMP		**Related Service Version: V_2.1**		
Request: eRoamingChargeDetailRecord		**Response: eRoamingAcknowledgement (see 4.2.6)**		
Direction: **EMP MUST receive message from HBS**		**Implementation:** **mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
Identification	IdentificationType (see 5.1.2)	Authentication data.	M	
ChargingStart	Date/Time	The date and time at which the charging process started.	O	
ChargingEnd	Date/Time	The date and time at which the charging process stopped.	O	
SessionStart	Date/Time	The date and time at which the session started, e.g. swipe of RFID or cable connected.	M	
SessionEnd	Date/Time	The date and time at which the session ended. E. g. Swipe of RFID or Cable disconnected	M	
MeterValueStart	Decimal (,3)	The starting meter value in kWh.	O	
MeterValueEnd	Decimal (,3)	The ending meter value in kWh.	O	
MeterValueInBetween	List (MeterValue(Decimal(,3)))	List of meter values that may have been taken in between (kWh).	O	
ConsumedEnergy	Decimal (,3)	The difference between MeterValueEnd and MeterValueStart in kWh.	O	
MeteringSignature	String	Meta data	O	200
HubOperatorID	OperatorIDType (see 5.3.4)	Hub operator	O	

Messages

HBS → EMP		**Related Service Version: V_2.1**		
Request: eRoamingChargeDetailRecord		**Response: eRoamingAcknowledgement (see 4.2.6)**		
Direction: **EMP MUST receive message from HBS**		**Implementation:** **mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
HubProviderID	ProviderIDType (see 5.3.3)	Hub provider	O	

4.2.2 eRoamingAuthorizeStart

eRoamingAuthorizeStart is a message to request an authorization for starting a charging process.

Please note:
This message describes the request which hast to be answered with the eRoamingAuthorizationStart response and is only mandatory for online EMPs.

HBS → EMP		**Related Service Version: V_2.1**		
Request: eRoamingAuthorizeStart		**Response: eRoamingAuthorizationStart (see 4.1.7)**		
Direction: **EMP MUST receive message from HBS**		**Implementation:** **mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process	O	
CPOPartnerSessionID	String	Optional field containing the session id assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250

Messages

HBS → EMP		**Related Service Version: V_2.1**		
Request: eRoamingAuthorizeStart		**Response: eRoamingAuthorizationStart (see 4.1.7)**		
Direction: **EMP MUST receive message from HBS**		**Implementation:** **mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
EMPPartnerSessionID	String	Optional field containing the session id assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
OperatorID	OperatorIDType (see 5.3.4)	The OperatorID is defined by Hubject and is used to identify the CPO.	M	
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	O	
Identification	IdentificationType (see 5.1.2)	Authentication data used to authorize the user or car.	M	
PartnerProductID	ProductIDType (see 5.2.12)	A pricing product name (for identifying a tariff) that must be unique	O	

4.2.3 eRoamingAuthorizeStop

eRoamingAuthorizeStop is a message to request an authorization for stopping a charging process.

HBS → EMP		**Related Service Version: V_2.1**		
Request: eRoamingAuthorizeStop		**Response: eRoamingAuthorizationStop (see 4.1.8)**		
Direction: **EMP MUST receive message from HBS**		**Implementation:** **mandatory**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process.	M	
CPOPartnerSessionID	String	Optional field containing the session id assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
EMPPartnerSessionID	String	Optional field containing the session id assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
OperatorID	OperatorIDType (see 5.3.4)	The OperatorID is defined by Hubject and is used to identify the CPO.	M	
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	O	
Identification	IdentificationType (see 5.1.2)	Authentication data used to authorize the user or car.	M	

4.2.4 eRoamingEVSEData

eRoamingEVSEData is sent in response to eRoamingPullEVSEData requests.

Please note:
This message describes the response which has to be received as response to the eRoamingPullEVSEData request.

HBS → EMP		**Related Service Version: V_2.2**	
Request: eRoamingPullEVSEData (see 4.1.2)		**Response: eRoamingEVSEData**	
Direction: **EMP MUST receive message from HBS**		**Implementation:** **mandatory**	
Name	**Data Type**	**Description**	**M/O**
EVSEData	List (OperatorEVSEDataType) (see 5.1.16)	A list of EVSE data blocks that are each assigned to a certain operator.	M
StatusCode	StatusCodeType (see 5.1.1)	This can be used e.g. for failure messages or further information regarding the result.	O

4.2.5 eRoamingEVSEStatus

eRoamingEVSEStatus is sent in response to eRoamingPullEVSEStatus requests.

Please note:
This message describes the response which will be received as response to the eRoamingPullEVSEStatus request.

HBS → EMP		**Related Service Version: V_2.1**	
Request: eRoamingPullEVSEStatus (see4.1.3)		**Response: eRoamingEVSEStatus**	
Direction: **EMP MUST receive message from HBS**		**Implementation:** **mandatory**	
Name	**Data Type**	**Description**	**M/O**
EVSEStatuses	List (OperatorEVSEStatusType) (see 5.1.18)	A list of EVSE status blocks that are each assigned to a certain operator.	M
StatusCode	StatusCodeType (see 5.1.1)	This can be used e.g. for failure messages or further information regarding the result.	O

4.2.6 eRoamingAcknowledgement

Acknowledgement is a message that is sent in response to several requests.

HBS → EMP				
Name	**Data Type**	**Description**	**M/O**	**Field length**
Result	Boolean	If result is true, the message was received and the respective operation was performed successfully. If result is false, the message was received and the respective operation was not performed successfully.	M	
StatusCode	StatusCodeType (see 5.1.1)	Structured status details. This can be used e.g. for failure messages or further information regarding the result.	M	
SessionID	SessionIDType (see 5.3.8)	Represents the service process. In some cases the current SessionID is returned to the service requestor in this field. In case of a remote transaction the Acknowledgement MUST include the corresponding session ID.	O	
CPOPartnerSessionID	String	Optional field containing the session id assigned by the CPO to the related operation.	O	250
EMPPartnerSessionID	String	Optional field containing the session id assigned by an EMP to the related operation.	O	250

Messages

4.3 Optional messages to be sent by EMPs

4.3.1 eRoamingPullEVSEStatusByID

eRoamingPullEVSEStatusByID is a message that is sent in order to request the EVSE status data for specific EVSE IDs.

EMP → HBS		**Related Service Version: V_2.1**	
Request: eRoamingPullEVSEStatusByID		**Response: eRoamingEVSEStatusByID (see 4.4.1)**	
Direction: EMP MAY send message to HBS		**Implementation: optional**	
Name	**Data Type**	**Description**	**M/O**
ProviderID	ProviderIDType (see 5.3.3)	Identifies the provider	M
EvseID	List (EvseIDType) (see 5.3.2)	The list MUST not contain more than 100 EvseIDs	M

4.3.2 eRoamingAuthorizeRemoteReservationStart

eRoamingAuthorizeRemoteReservationStart is a message to request a reservation of a charging spot.

EMP → HBS		**Related Service Version: V_1.1**		
Request: eRoamingAuthorizeRemoteReservationStart		**Response: eRoamingAcknowledgement (see 4.2.6)**		
Direction: EMP MAY send message to HBS		**Implementation: optional**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process.	O	

Messages

EMP → HBS		**Related Service Version: V_1.1**		
Request: **eRoamingAuthorizeRemoteReservationStart**		**Response:** **eRoamingAcknowledgement (see 4.2.6)**		
Direction: **EMP MAY send message to HBS**		**Implementation:** **optional**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
CPOPartnerSessionID	String	Optional field containing the session id assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
EMPPartnerSessionID	String	Optional field containing the session id assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
ProviderID	ProviderIDType (see 5.3.3)	The ProviderID is defined by Hubject and is used to identify the EMP.	M	
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	M	
Identification	IdentificationType (see 5.1.2)	Authentication data.	M	
PartnerProductID	ProductIDType (see 5.2.12)	A pricing product name (for identifying a tariff) that must be unique	O	
Duration	Integer	Duration of reservation in minutes.	O	2

Messages

4.3.3 eRoamingAuthorizeRemoteReservationStop

eRoamingAuthorizeRemoteReservationStop is a message to request the end of a reservation of a charging spot.

EMP → HBS		**Related Service Version: V_1.1**		
Request: **eRoamingAuthorizeRemoteReservationStop**		**Response:** **eRoamingAcknowledgement (see 4.2.6)**		
Direction: **EMP MAY send message to HBS**		**Implementation:** **optional**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
SessionID	SessionIDType (see 5.3.8)	The Hubject SessionID that identifies the process.	M	
CPOPartnerSessionID	String	Optional field containing the session id assigned by the CPO to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
EMPPartnerSessionID	String	Optional field containing the session id assigned by an EMP to the related operation. Partner systems can use this field to link their own session handling to HBS processes.	O	250
ProviderID	ProviderIDType (see 5.3.3)	The ProviderID is defined by Hubject and is used to identify the operating provider.	M	
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	M	

Messages

4.3.4 eRoamingPullEVSEStatusByOperatorID

eRoamingPullEVSEStatusByOperatorID is a message that is sent in order to request the EVSE status data for specific OperatorsIDs (i.e. CPO(s) specific EVSE status data).

EMP → HBS		**Related Service Version: V_2.1**	
Request: eRoamingPullEVSEStatusByOperatorID		**Response: eRoamingEVSEStatusByOperatorID (see 4.4.12)**	
Direction: EMP MAY send message to HBS		**Implementation: optional**	
Name	**Data Type**	**Description**	**M/O**
ProviderID	ProviderIDType (see 5.3.3)	Identifies the provider	M
OperatorID	OperatorIDType (see 5.3.4)	A list of Operator Ids in ISO or DIN standard to download only EVSE's of one or more operators	M

4.3.5 eRoamingPullPricingProductData

eRoamingPullPricingProductData is a message that is sent in order to request the download of pricing data available in the HBS for an EMP.

EMP → HBS		**Related Service Version: V_1.0**	
Request: eRoamingPullPricingProductData		**Response: eRoamingPricingProductData (see 4.2.43)**	
Direction: EMP MAY send message to HBS		**Implementation: optional**	
Recommended frequency: on demand			
Name	**Data Type**	**Description**	**M/O**
LastCall	Date/Time	In case that this field is set, Hubject does not return the entire set of currently valid pricing products data but just the changes that have taken places since the last call date/time value.	O
OperatorIDs	OperatorIDType (see 5.3.4)	A list of Operator Ids in ISO or DIN standard to download pricing data pushed by one or more operators.	M

4.3.6 eRoamingPullEVSEPricing

eRoamingPullEVSEPricing is a message that is sent in order to request the download of (i.e.pull) location/EVSE-specific pricing data uploaded by CPOs for the requesting EMP.

EMP → HBS	**Related Service Version: V_1.0**
Request: eRoamingPullEVSEPricing	**Response: eRoamingEVSEPricing (see 4.2.44)**
Direction: EMP MAY send message to HBS	**Implementation: optional**
Recommended frequency: on demand	

Name	Data Type	Description	M/O
ProviderID	ProviderIDType (see 5.3.3)	Identifies the provider requesting the data pull	M
LastCall	Date/Time	In case that this field is set, Hubject does not return the entire set of currently valid EVSE pricing data but just the changes that have taken place since the last call date/time value.	O
OperatorIDs	OperatorIDType (see 5.3.4)	A list of Operator Ids in ISO or DIN standard to download EVSE pricing data pushed by one or more operators.	M

4.4 Optional messages to be received by EMPs

4.4.1 eRoamingEVSEStatusByID

eRoamingEVSEStatusById is sent in response to eRoamingPullEVSEStatusById requests.

Please note:
This message describes the response which has to be sent in return to the eRoamingPullEVSEStatusByID request.

HBS → EMP		**Related Service Version: V_2.1**		
Request: **eRoamingPullEVSEStatusByID (see 4.3.1)**		**Response:** **eRoamingEVSEStatusByID**		
Direction: **EMP MAY receive message from HBS**		**Implementation:** **optional**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
EVSEStatusRecords	List(EVSEStatusRecord) (see 5.1.19)	A list of EVSE status records	O	
StatusCode	StatusCodeType (see 5.1.1)	This can be used e.g. for failure messages or further information regarding the result.	O	

Messages

4.4.2 eRoamingEVSEStatusByOperatorID

eRoamingEVSEStatusByOperatorID is sent in response to eRoamingPullEVSEStatusByOperatorID requests.

Please note:
This message describes the response which has to be sent in reply to the eRoamingPullEVSEStatusByOperatorID request.

HBS → EMP		**Related Service Version: V_2.1**		
Request: **eRoamingPullEVSEStatusByOperatorID (see 4.3.14)**		**Response:** **eRoamingEVSEStatusByOperatorID**		
Direction: **EMP MAY receive message from HBS**		**Implementation:** **optional**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
EVSEStatusRecords	List(EVSEStatusRecord) (see 5.1.19)	A list of EVSE status records	O	
StatusCode	StatusCodeType (see 5.1.1)	This can be used e.g. for failure messages or further information regarding the result.	O	

Messages

4.4.3 eRoamingPricingProductData

eRoamingPricingProductData is sent in response to eRoamingPullPricingProductData requests.

Please note:
This message describes the response which has to be sent in reply to the eRoamingPullPricingProductData request.

HBS → EMP		**Related Service Version: V_1.0**		
Request: **eRoamingPullPricingProductData (see 4.3.15)**		**Response:** **eRoamingPricingProductData**		
Direction: **EMP MAY receive message from HBS**		**Implementation:** **optional**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
OperatorPricingProducts	List(PricingProductDataType) (see 5.1.22)	List of pricing products offered by operators for a specific provider	M	
StatusCode	StatusCodeType (see 5.1.1)	This can be used e.g. for failure messages or further information regarding the result.	O	

Messages

4.4.4 eRoamingEVSEPricing

eRoamingEVSEPricing is sent by the HBS in response to eRoamingPullEVSEPricing requests.

Please note: This message describes the response which has to be sent in reply to the eRoamingPullEVSEPricing request.

HBS → EMP		**Related Service Version: V_1.0**		
Request: **eRoamingPullEVSEPricing (see 4.3.16)**		**Response:** **eRoamingEVSEPricing**		
Direction: **EMP MAY receive message from HBS**		**Implementation:** **optional**		
Name	**Data Type**	**Description**	**M/O**	**Field length**
OperatorEVSEPricing	List(OperatorEVSEPricingType) (see 5.1.32)	A list of EVSE pricing data blocks for specific operators	M	
StatusCode	StatusCodeType (see 5.1.1)	This can be used e.g. for failure messages or further information regarding the result.	O	

5 Data Types

5.1 Complex Data Types

Complex data types comprise a number of data fields that can also be complex types.

Best Practices
Best practices regarding datafields will be linked to the corresponding annex which contains detailed information.

5.1.1 StatusCodeType

The structure consists of a defined code, an optional functional description of the status, and optional additional information. It can be used e.g. to send error details or detailed reasons for a certain process or system behavior. The optional AdditionalInfo field can be used in order to provide further individual (non-standardized) information.

Name	Data Type	Description	M/O	Field length
Code	CodeType (see 5.2.10)	To be selected from valid range	M	
Description	String	Description	O	200
AdditionalInfo	String	More information can be provided here	O	1000

5.1.2 IdentificationType

Field Name	Field Type	Description	M/O
RFIDMifareFamilyIdentification	RFIDmifarefamilyIdentificationType (see 5.1.10)	Authentication data details. The data structure differs depending on the authentication technology. **Note:** (1) the option RFIDIdentification MUST not be used in the	M One of the five options MUST be provided
RFIDIdentification	RFIDIdentificationType (see 5.1.29)		
QRCodeIdentification	QRCodeIdentificationType (see 5.1.11)		
PlugAndChargeIdentification	PlugAndChargeIdentificationType (see 5.1.12)		

Data Types

Field Name	Field Type	Description	M/O
RemoteIdentification	RemoteIdentificationType (see 5.1.13)	eRoamingAuthorization process. For RFID Authorization, only the option *RFIDMifareFamilyIdentification* should be used in the respective eRoamingAuthorization messages. (2) For the Remote Authorization process, only the option *RemoteIdentification* MUST be used in the respective messages.	

5.1.3 EVSEMatchType

Name	Data Type	Description	M/O
EVSE	EVSEDataRecordType (see 5.1.17)	Charging point information.	M
Distance	Decimal (4,1)	Air distance to the requested position in km (non-routed).	M

5.1.4 GeoCoordinatesType

Field Name	Field Type	Description	M/O
Google	GeoCoordinatesGoogleType (see 5.1.5)	The data structure differs depending on the chosen geo coordinates format.	M One of the three options MUST be provided.
DecimalDegree	GeoCoordinatesDecimalDegreeType (see 5.1.6)		
DegreeMinuteSeconds	GeoCoordinatesDegreeMinuteSecondsType (see 5.1.7)		

Data Types

5.1.5 GeoCoordinatesGoogleType

Field Name	Field Type	Description	M/O
Coordinates	GeoCoordinatesGoogleFormatType (see 5.3.5)	Based on WGS84.	M

5.1.6 GeoCoordinatesDecimalDegreeType

Field Name	Field Type	Description	M/O
Longitude	GeoCoordinatesDecimalDegreeFormatType (see 5.3.6)	Based on WGS84.	M
Latitude	GeoCoordinatesDecimalDegreeFormatType (see 5.3.6)	Based on WGS84.	M

5.1.7 GeoCoordinatesDegreeMinuteSecondsType

Field Name	Field Type	Description	M/O
Longitude	GeoCoordinatesDegreeMinuteSecondsFormatType (see 5.3.7)	Based on WGS84.	M
Latitude	GeoCoordinatesDegreeMinuteSecondsFormatType (see 5.3.7)	Based on WGS84.	M

5.1.8 ProviderAuthenticationDataType

Field Name	Field Type	Description	M/O
ProviderID	ProviderIDType (see 5.3.3)	The EMP whose data records are listed below.	M
AuthenticationDataRecord	AuthenticationDataRecordType (see 5.1.9)		M 0..n

5.1.9 AuthenticationDataRecordType

Field Name	Field Type	Description	M/O
Identification	IdentificationType	Authentication data.	M

Data Types

Field Name	Field Type	Description	M/O
	(see 5.1.2)		

5.1.10 RFIDmifarefamilyIdentificationType

Name	Data Type	Description	M/O	Field length
UID	UIDType (see 5.3.10)	The UID from the RFID-Card. It should be read from left to right using big-endian format. Hubject will automatically convert all characters from lower case to upper case.	M	50

5.1.11 QRCodeIdentificationType

Name	Data Type	Description	M/O	Field length
EvcoID	EvcoIDType (see 5.3.1)	Contract identifier Hubject will automatically convert all characters from lower case to upper case.	M	
PIN	String	According to different processes, the PIN is transferred as encrypted hash or in clear text.	O One or none of the options can be provided.	20
HashedPIN	HashType (see 5.1.20)			

5.1.12 PlugAndChargeIdentificationType

Field Name	Field Type	Description	M/O
EvcoID	EvcoIDType (see 5.3.1)	Contract identifier.	M

5.1.13 RemoteIdentificationType

Field Name	Field Type	Description	M/O

Data Types

Field Name	Field Type	Description	M/O
EvcoID	EvcoIDType (see 5.3.1)	Contract identifier Hubject will automatically convert all characters from lower case to upper case.	M

5.1.14 AddressIso19773Type

Name	Data Type	M/O	Field length
Country	CountryCodeType (see 5.2.11)	M	
City	String	M	1-50
Street	String	M	2-100
PostalCode	String	O	10
HouseNum	String	O	10
Floor	String	O	5
Region	String	O	50
Timezone	TimezoneType (see 5.3.14)	O	

5.1.15 SearchCenterType

Name	Data Type	Description	M/O
GeoCoordinates	GeoCoordinatesType	(see 5.1.4)	M
Radius	Decimal (4,1)	Radius in km around the position that is defined by the geo coordinates.	M

5.1.16 OperatorEVSEDataType

Name	Data Type	Description	M/O	Field length
OperatorID	OperatorIDType (see 5.3.4)	The provider whose data records are listed below.	M	
OperatorName	String	Free text for operator	O	100
EVSEDataRecord	EVSEDataRecordType (see 5.1.17)	EVSE entries	M 0..n	

5.1.17 EVSEDataRecordType

Name	Data Type	Description	M/O	Field length
deltaType (attribute)	One of: ▪ update ▪ insert ▪ delete	In case that the operation "PullEVSEData" is performed with the parameter "LastCall", Hubject assigns this attribute to every response EVSE record in order to return the changes compared to the last call.	O	
lastUpdate	Date/Time	The attribute indicates the date and time of the last update of the record. Hubject assigns this attribute to every response EVSE record.	O	
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	M	
ChargingPoolID	ChargingPoolIDType (see 5.3.13)	Evses may be grouped by using a charging pool id	O	
ChargingStationId	String	The ID that identifies the charging station.	O	50

Data Types

Name	Data Type	Description	M/O	Field length
ChargingStationName	String	Name of the charging station.	O	50
EnChargingStationName	String	Name of the charging station in English.	O	50
Address	AddressIso19773Type (see 5.1.14)	Location of the charging station.	M	
GeoCoordinates	GeoCoordinatesType (see 5.1.4)	Location of the charging station.	M	
Plugs	List (PlugType) (see 5.2.3)	List of plugs that are supported.	M	
ChargingFacilities	List (ChargingFacilityType) (see 5.1.21)	List of facilities that are supported.	O	
ChargingModes	List (ChargingModeType) (see 5.2.4)	List of charging modes that are supported.	O	
AuthenticationModes	List (AuthenticationModeType) (see 5.2.5)	List of authentication modes that are supported.	M	
MaxCapacity	Integer	Maximum capacity in kWh.	O	
PaymentOptions	List (PaymentOptionType) (see 5.2.6)	List of payment options that are supported.	O	
ValueAddedServices	List ValueAddedServiceType (see 5.2.13)	List of value added services that are supported.	M	
Accessibility	AccessibilityType (see 5.2.2)	Specifies how the charging station can be accessed.	M	
HotlinePhoneNum	PhoneNumberType (see 5.3.9)	Phone number of a hotline of the charging station operator.	M	

Data Types

Name	Data Type	Description	M/O	Field length
AdditionalInfo	InfoTextType (see 5.1.30)	Optional information.	O	
GeoChargingPointEntrance	GeoCoordinatesType (see 5.1.4)	In case that the charging spot is part of a bigger facility (e.g. parking place), this attribute specifies the facilities entrance coordinates.	O	
isOpen24Hours	Boolean	Set in case the charging spot is open 24 hours.	M	
OpeningTimes	OpeningTimesType (see 5.1.31)	Opening time in case that the charging station cannot be accessed around the clock.	O	
HubOperatorID	OperatorIDType (see 5.3.4)	Hub operator.	O	
ClearinghouseID	String	Identification of the corresponding clearing house in the event that roaming between different clearing houses must be processed in the future.	O	20
IsHubjectCompatible	Boolean	Is eRoaming via intercharge at this charging station possible? If set to "false" the charge spot will not be started/stopped remotely via Hubject.	M	
DynamicInfoAvailable	Enumeration	Values: true / false / auto This attribute indicates whether a CPO provides (dynamic) EVSE Status info in addition to the (static) EVSE Data for this EVSERecord. Value auto is set to true by Hubject if the operator offers Hubject EVSEStatus data.	M	

5.1.18 OperatorEVSEStatusType

Name	Data Type	Description	M/O	Field length
OperatorID	OperatorIDType (see 5.3.4)	The provider whose status records are listed below.	M	
OperatorName	String	Operator name	O	100
EVSEStatusRecord	EVSEStatusRecordType (see 5.1.19)	EVSEStatus list	M 0..n	

5.1.19 EVSEStatusRecordType

Name	Data Type	Description	M/O
EvseID	EvseIDType (see 5.3.2)	The ID that identifies the charging spot.	M
EVSEStatus	EVSEStatusType (see 5.2.9)	The status of the charging spot.	M

5.1.20 HashType

Name	Data Type	Description	M/O	Field length
Value	HashValueType (see 5.3.11)	Hash value.	M	
Function	HashFunctionType	Function that was used to generate the hash value.	M	
Salt	String	In case that a salt value was used to generate the hash value (e.g. salted SHA-1 hash) the salt can be provided in this field. Where the Hash Function is Bcrypt, this value is undefined.	O	100

Data Types

5.1.21 ChargingFacilityType

Name	Data Type	Description	M/O	Field length
PowerType	List (PowerType) (see 5.2.14)	Charging Facility power type (e.g. AC or DC)	O	
Voltage	Integer	Voltage of the Charging Facility	O	
Amperage	Integer	Amperage of the Charging Facility	O	
Power	Decimal	Charging Facility power in kW	O	

5.1.22 PricingProductDataType

Name	Data Type	Description	M/O	Field length
OperatorID	OperatorIDType (see 5.3.4)	The Operator whose data records are listed below	M	
OperatorName	String	Free text for operator Name	O	
ProviderID	ProviderIDType (see 5.3.3)	The EMP for whom the pricing data is applicable. In case the data is to be made available for all EMPs (e.g. for Offer-to-All prices), the asterix character (*) can be set as the value in this field.	M	
PricingDefaultPrice	Decimal	A default price for pricing sessions at undefined EVSEs	M	
PricingDefaultPriceCurrency	CurrencyIDType (see 5.2.17)	Currency for default prices	M	
PricingDefaultReferenceUnit	ReferenceUnitType (see 5.2.15)	Default Reference Unit in time or kWh	M	
PricingProductDataRecords	PricingProductDataRecordType (see 5.1.23)	A list of pricing products	M	0…n

5.1.23 PricingProductDataRecordType

Name	Data Type	Description	M/O	Field length
ProductID	ProductIDType (see 5.2.12)	A pricing product name (for identifying a tariff) that must be unique	M	50
ReferenceUnit	ReferenceUnitType (see 5.2.15)	Reference unit in time or kWh	M	
PricePerReferenceUnit	Decimal	A price per the selected reference unit	M	
ProductPriceCurrency	CurrencyIDType (see 5.2.16)	The currency of the defined price	M	
MaximumProductChargingPower	Decimal	A value in kWh	M	
IsValid24hours	Boolean (TRUE or FALSE)	Set to TRUE if the respective pricing product is applicable 24 hours a day. If FALSE, the respective applicability times should be provided in the field “ProductAvailabilityTimes”.	M	
ProductAvailabilityTimes	ProductAvailabilityTimesType (see 5.1.26)	A list indicating when the pricing product is applicable	M	
AdditionalReferences	AdditionalReferencesType (see 5.1.27)	A list of additional reference units and their respective prices	O	0…n

5.1.24 EVSEPricingType

Name	Data Type	Description	M/O	Field length
EvseID	EvseIDType (see 5.3.2)	The EvseID of an EVSE for which the defined pricing products are applicable	M	
ProviderID	ProviderIDType (see 5.3.3)	EMP for whom the data below is pushed. In case the data is to be made available for all EMPs (e.g. for Offer-to-All prices), the asterix character (*) can be set as the value in this field.	M	
EvseIDProductList	EvseIDProductListType (see 5.1.25)	A list of pricing products applicable per EvseID	M	1…n

5.1.25 EvseIDProductListType

Name	Data Type	Description	M/O	Field length
ProductID	String	The product name of the applicable pricing product	M	50

5.1.26 ProductAvailabilityTimesType

Name	Data Type	Description	M/O	Field length
Periods	PeriodType (see 5.1.28)	The starting and end time for pricing product applicability in the specified period	M	
On	enum (Everyday, Workdays, Weekend, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday)	Day values to be used in specifying periods on which the product is available	M	

5.1.27 AdditionalReferencesType

Name	Data Type	Description	M/O	Field length
AdditionalReference	AdditionalReferenceType (see 5.2.16)	Additional pricing components to be considered in addition to the base pricing	M	
AdditionalReferenceUnit	ReferenceUnitType (see 5.2.15)	reference unit for the additional pricing components	M	
PricePerAdditionalReferenceUnit	Decimal	A price in the given currency	M	

5.1.28 PeriodType

Name	Data Type	Description	M/O	Field length
begin	String Pattern: "[0-9]{2}:[0-9]{2}"	The opening time	M	
end	String Pattern: "[0-9]{2}:[0-9]{2}"	The closing time	M	

5.1.29 RFIDIdentificationType

Name	Data Type	Description	M/O	Field length
UID	UIDType (see 5.3.10)	The UID from the RFID-Card. It should be read´from left to right using big-endian format. Hubject will automatically convert all characters from lower case to upper case	M	
EvcoID	EvcoIDType (see 5.3.1)	Contract identifier	O	
RFID	RFIDType (see 5.2.18)	The Type of the used RFID card like mifareclassic, desfire	M	
PrintedNumber	String	A number printed on a customer's card for manual authorization (e.q. via a call center)	O	150
ExpiryDate	Date/Time	Until when this card is valid. Should not be set if card does not have an expiration	O	

5.1.30 InfoTextType

Name	Data Type	Description	M/O	Field length
lang	LanguageCodeType (see 5.3.12)	The language in which the additional info text is provided	M	
value	String	The Additional Info text	M	

5.1.31 OpeningTimesType

Name	Data Type	Description	M/O	Field length
Periods	PeriodType (see 5.1.28)	The starting and end time for pricing product applicability in the specified period	M	
On	enum (Everyday, Workdays, Weekend, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday)	Day values to be used in specifying periods on which the product is available. Workdays = Monday – Friday Weekend = Saturday – Sunday	M	

5.1.32 OperatorEVSEPricingType

Name	Data Type	Description	M/O	Field length
OperatorID	OperatorIDType (see 5.3.4)	The provider whose status records are listed below.	M	
OperatorName	String	Operator name	O	100
EVSEPricing	EVSEPricingType (see 5.1.24)	List of EVSE pricings offered by the operator.	M 0..n	

5.2 Simple Specification Data Types

Specification types define a range of possible data values the data field that is assigned to the type can have.

5.2.1 AuthorizationStatusType

Option	Description
Authorized	User is authorized.
NotAuthorized	User is not authorized.

5.2.2 AccessibilityType

Option	Description
Free publicly accessible	Defined type of accessibility.
Restricted access	Defined type of accessibility.
Paying publicly accessible	Defined type of accessibility.
Unspecified	Defined type of accessibility.
Test Station	Defined type of accessibility.

5.2.3 PlugType

Option	Description
Small Paddle Inductive	Defined plug type.
Large Paddle Inductive	Defined plug type.
AVCON Connector	Defined plug type.
Tesla Connector	Defined plug type.
NEMA 5-20	Defined plug type.
Type E French Standard	CEE 7/5.
Type F Schuko	CEE 7/4.

Data Types

Option	Description
Type G British Standard	BS 1363.
Type J Swiss Standard	SEV 1011.
Type 1 Connector (Cable Attached)	Cable attached to IEC 62196-1 type 1, SAE J1772 connector.
Type 2 Outlet	IEC 62196-1 type 2.
Type 2 Connector (Cable Attached)	Cable attached to IEC 62196-1 type 2 connector.
Type 3 Outlet	IEC 62196-1 type 3.
IEC 60309 Single Phase	IEC 60309.
IEC 60309 Three Phase	IEC 60309.
CCS Combo 2 Plug (Cable Attached)	IEC 62196-3 CDV DC Combined Charging Connector DIN SPEC 70121 refers to ISO / IEC 15118-1 DIS, -2 DIS and 15118-3.
CCS Combo 1 Plug (Cable Attached)	IEC 62196-3 CDV DC Combined Charging Connector with IEC 62196-1 type 2 SAE J1772 connector.
CHAdeMO	DC CHAdeMO Connector.
Unspecified	Defined plug type.

5.2.4 ChargingModeType

Option	Description
Mode_1	IEC 61851-1.
Mode_2	IEC 61851-1.
Mode_3	IEC 61851-1.
Mode_4	IEC 61851-1.
CHAdeMO	CHAdeMo Specification.

5.2.5 AuthenticationModeType

Option	Description
NFC RFID Classic	Defined authentication.
NFC RFID DESFire	Defined authentication.
PnC	ISO/IEC 15118.
REMOTE	App, QR-Code, Phone.
Direct Payment	Remote use via direct payment. E.g. intercharge *direct*

5.2.6 PaymentOptionType

Option	Description
No Payment	Free.
Direct	e. g. Cash, Card, SMS, …
Contract	i. e. Subscription.

5.2.7 GeoCoordinatesResponseFormatType

Option	Description
Google	Based on WGS84.
DegreeMinuteSeconds	Based on WGS84.
DecimalDegree	Based on WGS84.

5.2.8 HashFunctionType

Option	Description
Bcrypt	Hash value is based on Bcrypt.

5.2.9 EVSEStatusType

Option	Description
Available	Charging Spot is available for charging.
Reserved	Charging Spot is reserved and not available for charging.
Occupied	Charging Spot is busy.
OutOfService	Charging Spot is out of service and not available for charging.
EVSENotFound	The requested EVSEID and EVSE status does not exist within the Hubject database.
Unknown	No status information available.

5.2.10 CodeType (list of error and status codes)

Option	Description	Area of usage
000	Success.	General codes
001	Hubject system error.	Internal system codes
002	Hubject database error.	Internal system codes
009	Data transaction error.	Internal system codes
017	Unauthorized Access.	Internal system codes
018	Inconsistent EVSEID.	Internal system codes
019	Inconsistent EvcoID.	Internal system codes
021	System error.	General codes
022	Data error.	General codes
101	QR Code Authentication failed – Invalid Credentials.	Authentication codes
102	RFID Authentication failed – invalid UID.	Authentication codes
103	RFID Authentication failed – card not readable.	Authentication codes
105	PLC Authentication failed - invalid EvcoID.	Authentication codes
106	No positive authentication response.	Authentication codes / Internal system codes
110	QR Code App Authentication failed – time out error.	Authentication codes
120	PLC (ISO/ IEC 15118) Authentication failed – invalid	Authentication codes

Data Types

Option	Description	Area of usage
	underlying EvcoID.	
121	PLC (ISO/ IEC 15118) Authentication failed – invalid certificate.	Authentication codes
122	PLC (ISO/ IEC 15118) Authentication failed – time out error.	Authentication codes
200	EvcoID locked.	Authentication codes
210	No valid contract.	Session codes
300	Partner not found.	Session codes
310	Partner did not respond.	Session codes
320	Service not available.	Session codes
400	Session is invalid.	Session codes
501	Communication to EVSE failed.	EVSE codes
510	No EV connected to EVSE.	EVSE codes
601	EVSE already reserved.	EVSE codes
602	EVSE already in use/ wrong token.	EVSE codes
603	Unknown EVSE ID.	EVSE codes
604	EVSE ID is not Hubject compatible.	EVSE codes
700	EVSE out of service.	EVSE codes

5.2.11 CountryCodeType

The CountryCodeType allows for Alpha-3 country codes only as of OICP 2.2.

For Alpha-3 (three-letter) country codes as defined in ISO 3166-1 see http://unstats.un.org/unsd/methods/m49/m49alpha.htm for the full code list.

Examples:

Option	Description
AUT	Austria
DEU	Germany
FRA	France
USA	United States

5.2.12 ProductIDType

The ProductIDType defines some standard values (see below). The type however also supports custom ProductIDs that can be specified by partners (as a string of 50 characters maximum length).

Option	Description
Standard Price	Standard price
AC1	Product for AC 1 Phase charging
AC3	Product for AC 3 Phase charging
DC	Product for DC charging
CustomProductID	There is no option “CustomProductID”, this sample option is meant to indicates that custom product ID specifications by partners (as a string of 50 characters maximum length) are allowed as well.

5.2.13 ValueAddedServiceType

Option	Description
Reservation	Can an EV driver reserve the charging sport via remote services?
DynamicPricing	Does the EVSE ID support dynamic pricing?
ParkingSensors	Is for this EVSE ID a dynamic status of the corresponding parking lot in front of the EVSE-ID available?
MaximumPowerCharging	Does the EVSE-ID offer a dynamic maximum power charging?
PredictiveChargePointUsage	Is for the EVSE-ID a predictive charge Point Usage available?
ChargingPlans	Does the EVSE-ID offer charging plans, e.g. As described in ISO15118-2?
None	There are no value added services available.

5.2.14 PowerType

Option	Description
AC_1_PHASE	Defined Charging Facility Power Type.
AC_3_PHASE	Defined Charging Facility Power Type.
DC	Defined Charging Facility Power Type.
Unspecified	Defined Charging Facility Power Type.

5.2.15 ReferenceUnitType

Option	Description
HOUR	Defined Reference Unit Type
KILOWATT_HOUR	Defined Reference Unit Type
MINUTE	Defined Reference Unit Type

5.2.16 AdditionalReferenceType

Option	Description
START FEE	Can be used in case a fixed fee is charged for the initiation of the charging session. This is a fee charged on top of the main base price defined in the field "*PricePerReferenceUnit*" for any particular pricing product
FIXED FEE	Can be used if a single price is charged irrespective of charging duration or energy consumption (for instance if all sessions are to be charged a single fixed fee). When used, the value set in the field "*PricePerReferenceUnit*" for the main base price of respective pricing product should be set to zero.
PARKING FEE	Can be used in case sessions are to be charged for both parking and charging. When used, it needs to be specified in the corresponding service offer on the HBS Portal when parking applies (e.g. from *session start* to *charging start* and *charging end* to *session end* or for the entire session duration, or *x*-minutes after *charging end*, etc)
MINIMUM FEE	Can be used in case there is a minimum fee to be paid for all charging sessions. When used, this implies that the eventual price to be paid cannot be less than this minimum fee but can however be a price above/greater than the minimum fee.
MAXIMUM FEE	Can be used in case there is a maximum fee to be charged for all charging sessions. When used, this implies that the eventual price to be paid cannot be more than this maximum fee but can however be a price below/lower than the maximum fee.

5.2.17 CurrencyIDType

The ProductPriceCurrencyType allows for the list of active codes of the official ISO 4217 currency names.

For the full list of active codes of the official ISO 4217 currencies, see: https://en.wikipedia.org/wiki/ISO_4217

Examples:

Option	Description
EUR	Euro
CHF	Swiss franc
CAD	Canadian Dollar
GBP	Pound sterling

5.2.18 RFIDType

Option	Description
mifareCls	Defined RFID Type
mifareDes	Defined RFID Type
calypso	Defined RFID Type
nfc	Defined RFID Type
mifareFamily	Defined RFID Type

5.3 Simple Restricted String Data Types

Restricted string types define a string that is restricted with respect to a certain regular expression.

5.3.1 EvcoIDType

A string that MUST be valid with respect to the following regular expression: ISO | DIN.

^(([A-Za-z]{2}\-?[A-Za-z0-9]{3}\-?C[A-Za-z0-9]{8}\-?[\d|A-Za-z])|
([A-Za-z]{2}[*|\-]?[A-Za-z0-9]{3}[*|\-]?[A-Za-z0-9]{6}[*|\-]?[\d|X]))$

The expression validates the string as EvcoID. It supports both definitions DIN SPEC 91286:2011-11 as well as ISO 15118-1.
In case the EvcoID is provided corresponding to ISO, the instance part must be eight characters long and must be provided with a prepended “C”. The optional separating character must be “-“.
In case the EvcoID is provided corresponding to DIN, the instance part must be six characters long. The optional separating character can either be “*” or “-“.

Examples ISO: “DE-8EO-CAet5e4XY-3”, “DE8EOCAet5e43X1”
Examples DIN: “DE*8EO*Aet5e4*3”, “DE-8EO-Aet5e4-3”, “DE8EOAet5e43”

Best Practices
For EMP’s using the ISO Standard of EvcoID’s, we recommend to use the following marked characters whithin the EvcoID: DE-ABC-C123456QQ-9

5.3.2 EvseIDType

A string that MUST be valid with respect to the following regular expression: ISO | DIN.

^(([A-Za-z]{2}*?[A-Za-z0-9]{3}*?E[A-Za-z0-9*]{1,30})|(\+?[0-9]{1,3}*[0-9]{3}*[0-9*]{1,32}))$

The expression validates the string as EvseID. It supports both definitions DIN SPEC 91286:2011-11 as well as ISO 15118-1.
In case the EvseID is provided corresponding to ISO, the country code must be provided as Alpha-2-Code (DIN EN ISO-3166-1) and the separator character “*” is optional. Furthermore the ID must provide an “E” after the OperatorID in order to identify the ID as ISO EvseID without doubt.
In case the EvseID is provided corresponding to DIN, the country code must be provided according to the international telecommunication numbering plan (ITU-T E.164:11/2010) and the separator character “*” is mandatory.

Examples ISO: “DE*AB7*E840*6487”, “DEAB7E8406487”
Example DIN: “+49*810*000*438”

5.3.3 ProviderIDType

A string that MUST be valid with respect to the following regular expression: ISO | DIN

^([A-Za-z]{2}\-?[A-Za-z0-9]{3}|[A-Za-z]{2}[*|-]?[A-Za-z0-9]{3})$

The expression validates the string as ProviderID including the preceding country code, which is part of EvcoID. It supports both definitions DIN SPEC 91286:2011-11 as well as ISO 15118-1.
In case the ProviderID is provided corresponding to ISO, the country code must be provided as Alpha-2-Code (DIN EN ISO-3166-1) and the separator character “-” is optional.

Examples ISO: “DE8EO”, “DE-8EO”
Examples DIN: “DE8EO”, “DE*8EO”, “DE-8EO”

5.3.4 OperatorIDType

A string that MUST be valid with respect to the following regular expression: ISO | DIN

^(([A-Za-z]{2}*?[A-Za-z0-9]{3})|(\+?[0-9]{1,3}*[0-9]{3}))$

The expression validates the string as OperatorID including the preceding country code, which is part of EvseID. It supports both definitions DIN SPEC 91286:2011-11 as well as ISO 15118-1.
In case the OperatorID is provided corresponding to ISO, the country code must be provided as Alpha-2-Code (DIN EN ISO-3166-1) and the separator character “*” is optional.
In case the OperatorID is provided corresponding to DIN, the country code must be provided according to the international telecommunication numbering plan (ITU-T E.164:11/2010) and the separator character “*” is mandatory.

Examples ISO: “DE*A36”, “DEA36”
Example DIN: “+49*536”

5.3.5 GeoCoordinatesGoogleFormatType

A string that MUST be valid with respect to the following regular expression:

^-?1?\d{1,2}\.\d{1,6}\s*\,?\s*-?1?\d{1,2}\.\d{1,6}$

The expression validates the string as geo coordinates with respect to the Google standard. The string contains latitude and longitude (in this sequence) separated by a space.

Example: “47.662249 9.360922”

5.3.6 GeoCoordinatesDecimalDegreeFormatType

A string that MUST be valid with respect to the following regular expression:

^-?1?\d{1,2}\.\d{1,6}$

The expression validates the string as a geo coordinate (longitude or latitude) with decimal degree syntax.

Examples: “9.360922”, “-21.568201”

5.3.7 GeoCoordinatesDegreeMinuteSecondsFormatType

A string that MUST be valid with respect to the following regular expression:

^-?1?\d{1,2}°[ ]?\d{1,2}'[ ]?\d{1,2}\.\d+’'$

The expression validates the string as a geo coordinate (longitude or latitude) consisting of degree, minutes, and seconds.

Examples: “9°21'39.32"”, “-21°34'23.16"

5.3.8 SessionIDType

A string that MUST be valid with respect to the following regular expression:

^[A-Za-z0-9]{8}(-[A-Za-z0-9]{4}){3}-[A-Za-z0-9]{12}$

The expression validates the string as a GUID.

Example: “b2688855-7f00-0002-6d8e-48d883f6abb6”

5.3.9 PhoneNumberType

^\+[0-9]{5,15}$

The expression validates the string as a telephone number starting with “+” and containing only numbers.

Example: “+0305132787”

5.3.10 UIDType

^([0-9A-F]{8,8}|[0-9A-F]{14,14}|[0-9A-F]{20,20})$

The expression validates the string as a unique RFID with a length of 8, 14 or 20 characters.

Examples: “AFFH1768”, “7568290FFF765F”

5.3.11 HashValueType

^[0-9A-Za-z]{10,100}$

The expression validates the string as a hash function result value with a length between 10 and 100 characters

Example: “a5ghdhf73h”

5.3.12 LanguageCodeType

^[a-z]{2,3}(?:-[A-Z]{2,3}(?:-[a-zA-Z]{4})?)?(?:-x-[a-zA-Z0-9]{1,8})?$

The expression validates the string as a language code as per ISO-639-1 or ISO-639-2/T (see https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes)

The LanguageCodeType is used in the AdditionalInfo field, which is part of the EvseDataRecordType (see 5.1.17).

5.3.13 ChargingPoolIDType

EVSEs may be grouped by using a charging pool id according to emi³ standard definition. The Evse Pool ID MUST match the following structure (the notation corresponds to the augmented Backus-Naur Form (ABNF) as defined in RFC5234):

<Evse Pool ID> = <Country Code> <S> <Spot Operator ID> <S> <ID Type> <Pool ID>
with:
<Country Code> = 2 ALPHA ; two character country code according to ISO-3166-1 (Alpha-2-Code).

Data Types

<Spot Operator ID> = 3 (ALPHA / DIGIT); three alphanumeric characters.

<ID Type> = “P”; one character “P” indicating that this ID represents a “Pool”.

<Pool Instance> = (ALPHA / DIGIT) 1 * 30 (1*(ALPHA / DIGIT) [/ <S>]); between 1 and 31sequence of alphanumeric characters, including additional optional separators. Starts with alphanumeric character referring to a specific Pool in EVSE Operator data system.

- ALPHA = %x41-5A / %x61-7A; according to RFC 5234 (7-Bit ASCII).
- DIGIT = %x30-39; according to RFC 5234 (7-Bit ASCII).
- <S> = *1 (“*”); optional separator

An example for a valid Evse Pool ID is “IT*123*P456*AB789” with :

- “IT” indicating Italy,
- “123” representing a particular Spot Operator,
- “P” indicating the Pool, and
- “456*AB789” representing one of its POOLs.

NOTE: In contrast to the eMA ID, no check digit is specified for the Evse Pool ID in this document. Alpha characters SHALL be interpreted case insensitively. emi³ strongly recommends that implementations SHOULD

- use the separator between Country Code and Spot Operator ID
- use the separator between Spot Operator ID and ID type

This leads to the following regular expression:

([A-Za-z]{2}*?[A-Za-z0-9]{3}*?P[A-Za-z0-9*]{1,30})

This regular expression is similar to that of the ISO EvseIDType but E is replaced with P to indicate a pool.

5.3.14 TimezoneType

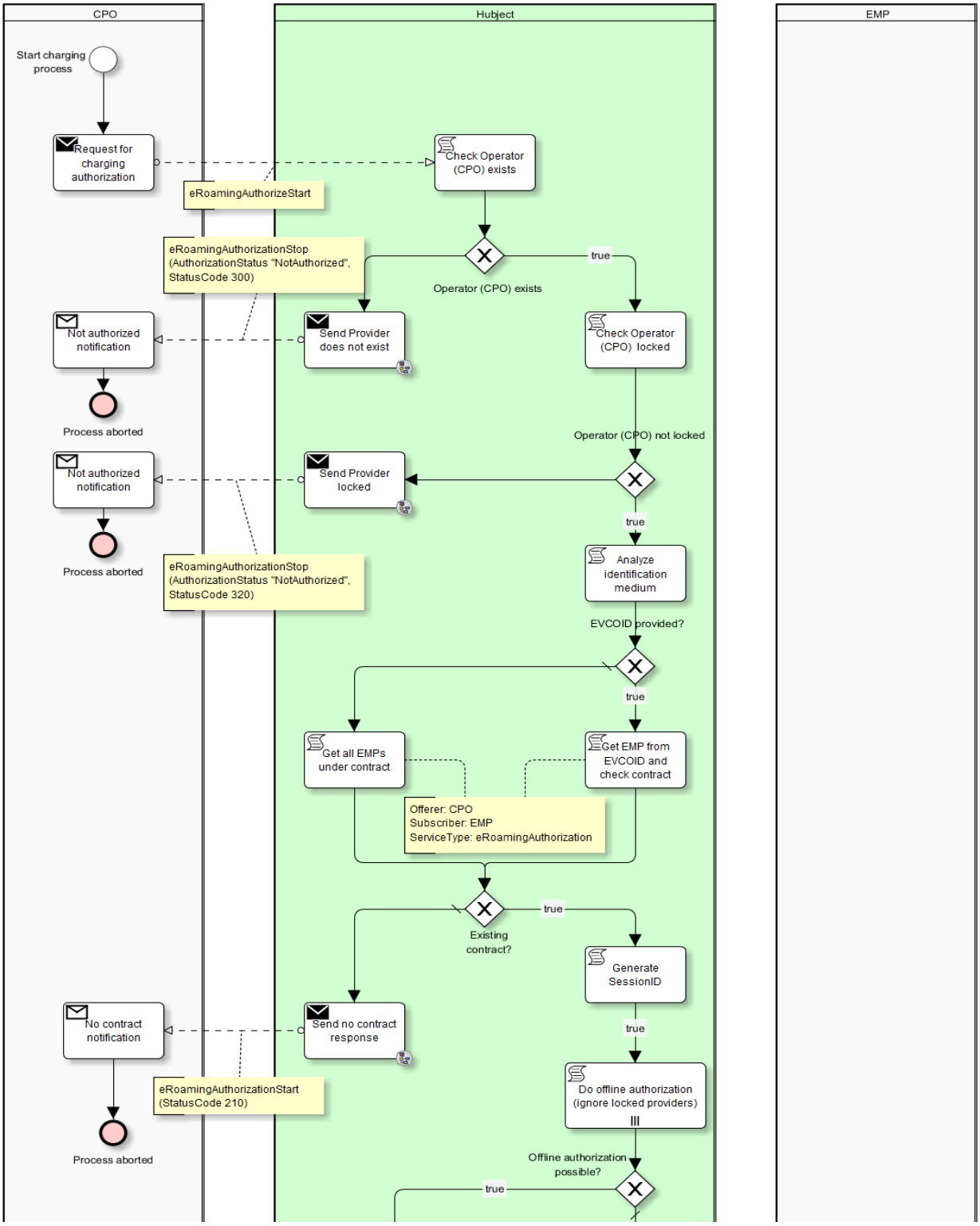
[U][T][C][+,-][0-9][0-9][:][0-9][0-9]

The expression validates a string as a Time zone with UTC offset.

Examples:
UTC+01:00
UTC-05:00

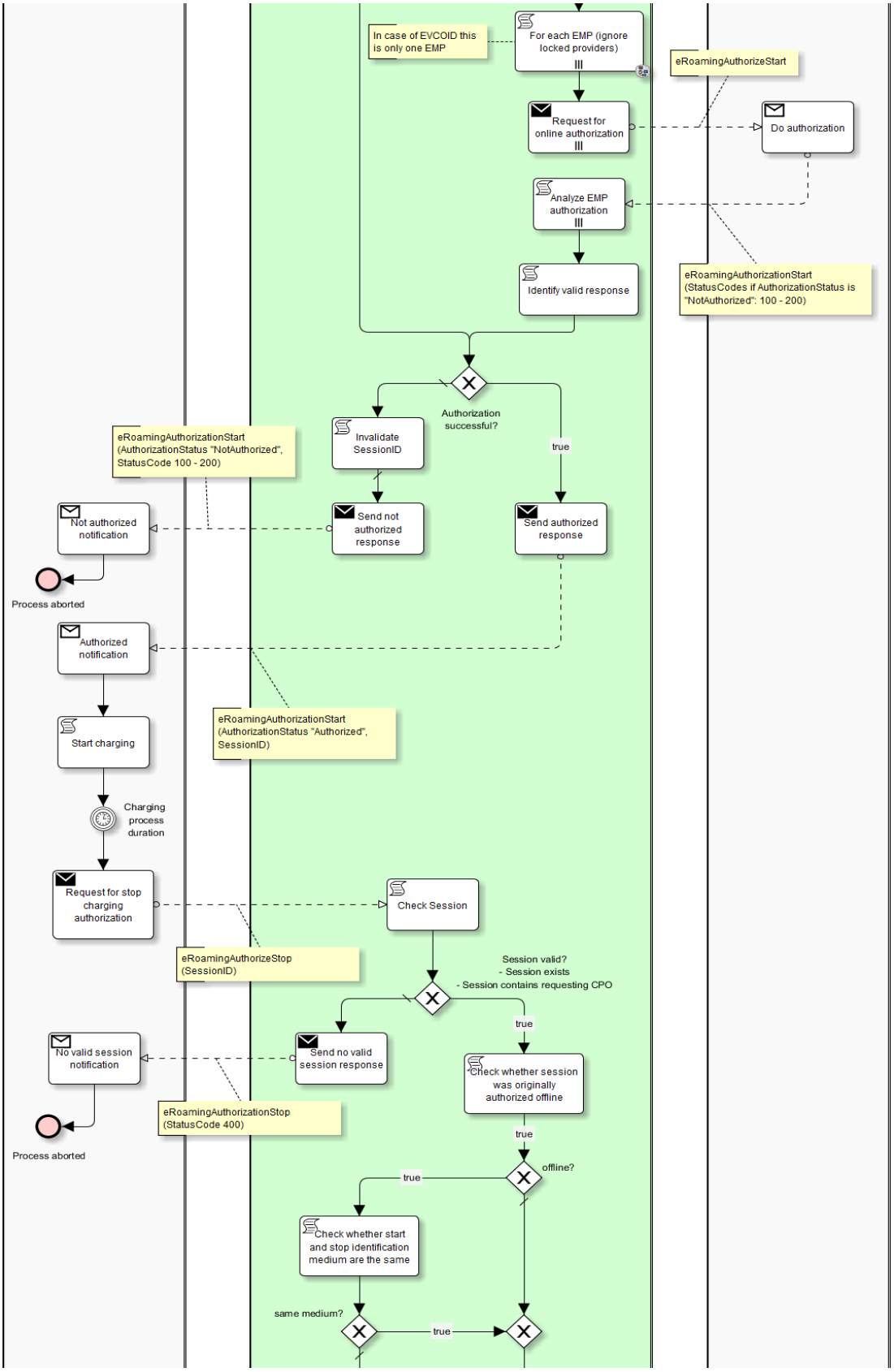
6 Appendix

6.1 Business Process Diagram eRoamingAuthorization



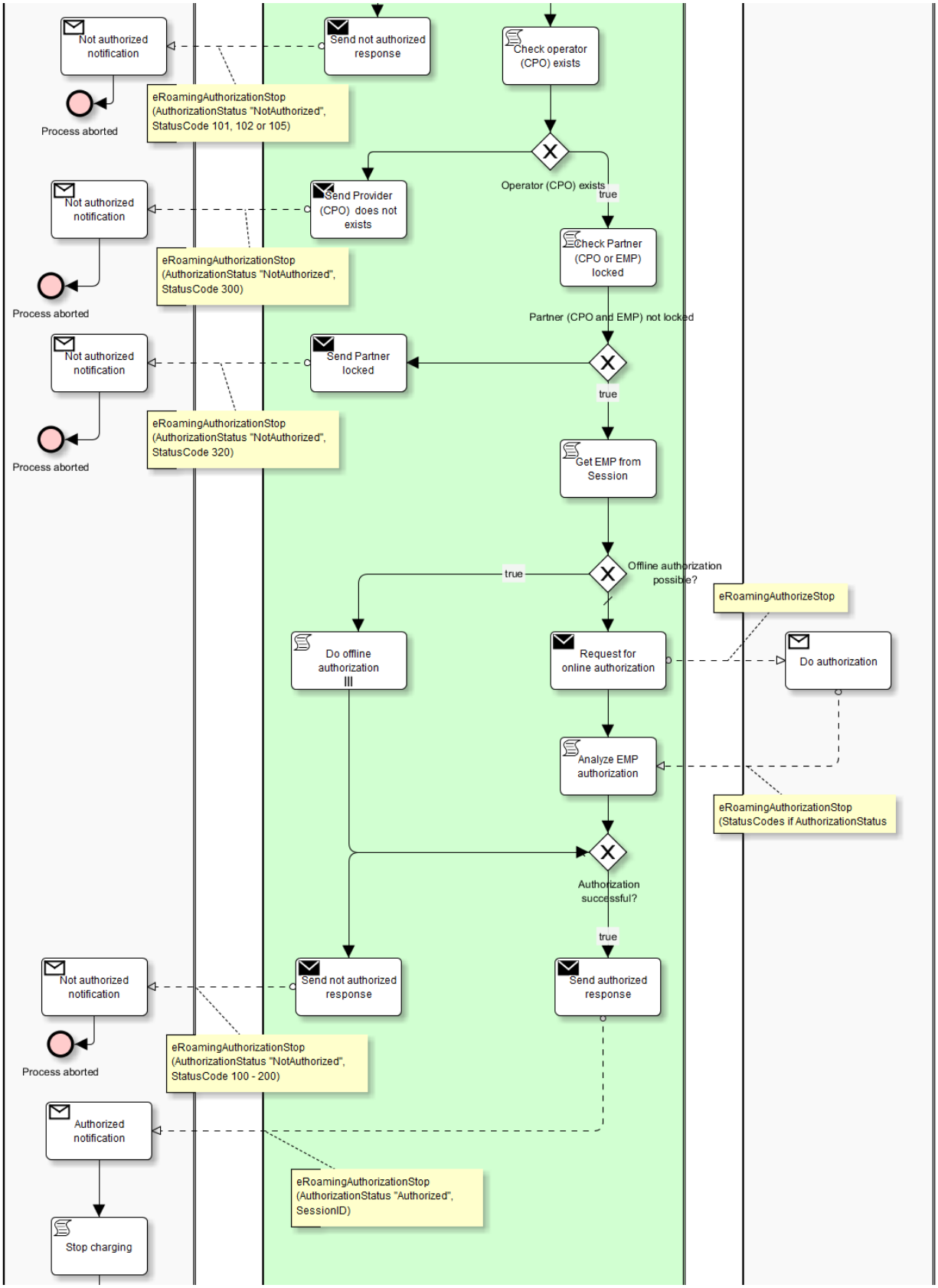
Appendix

Business Process Diagram eRoamingAuthorization



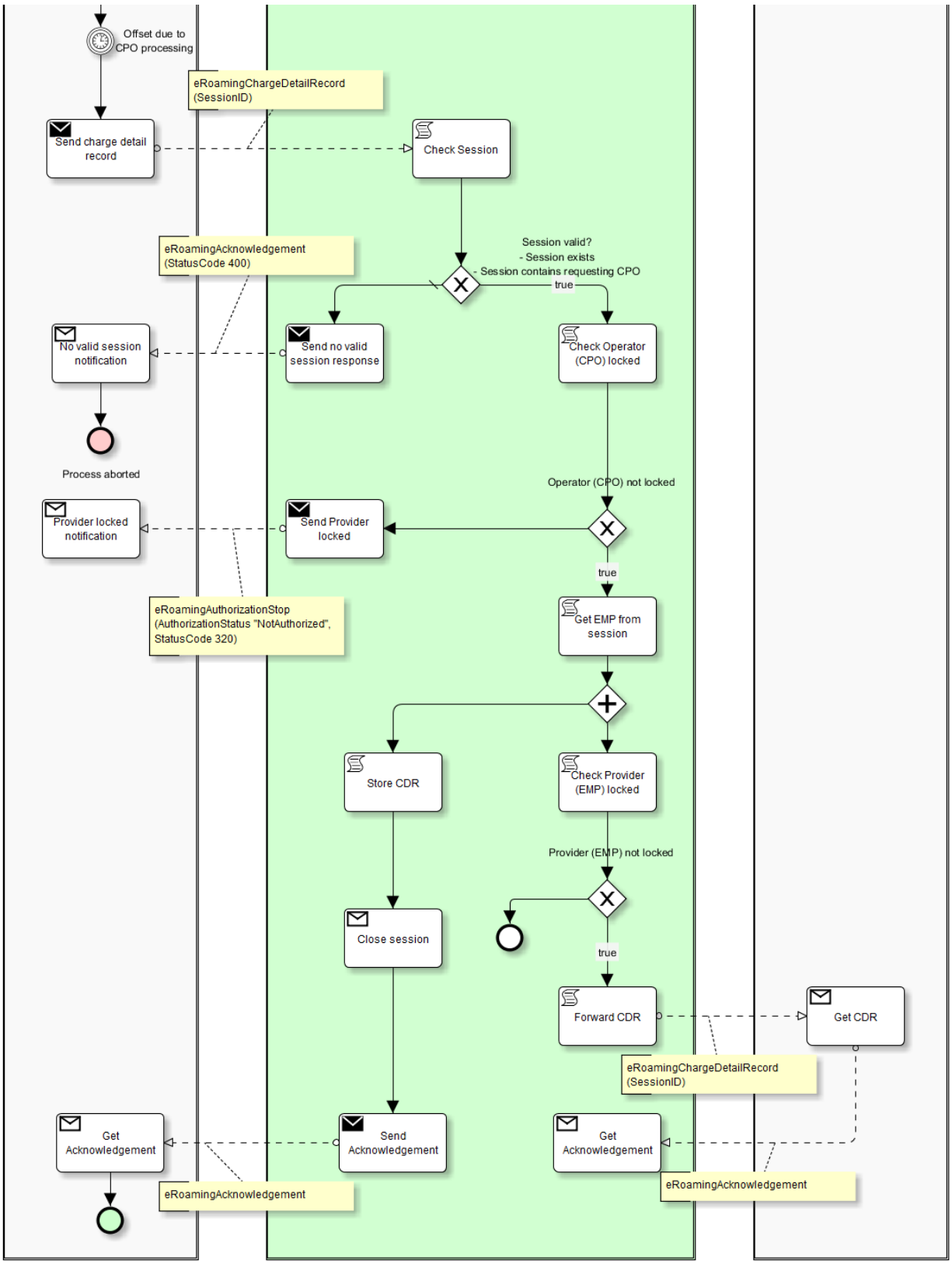
Appendix

Business Process Diagram eRoamingAuthorization

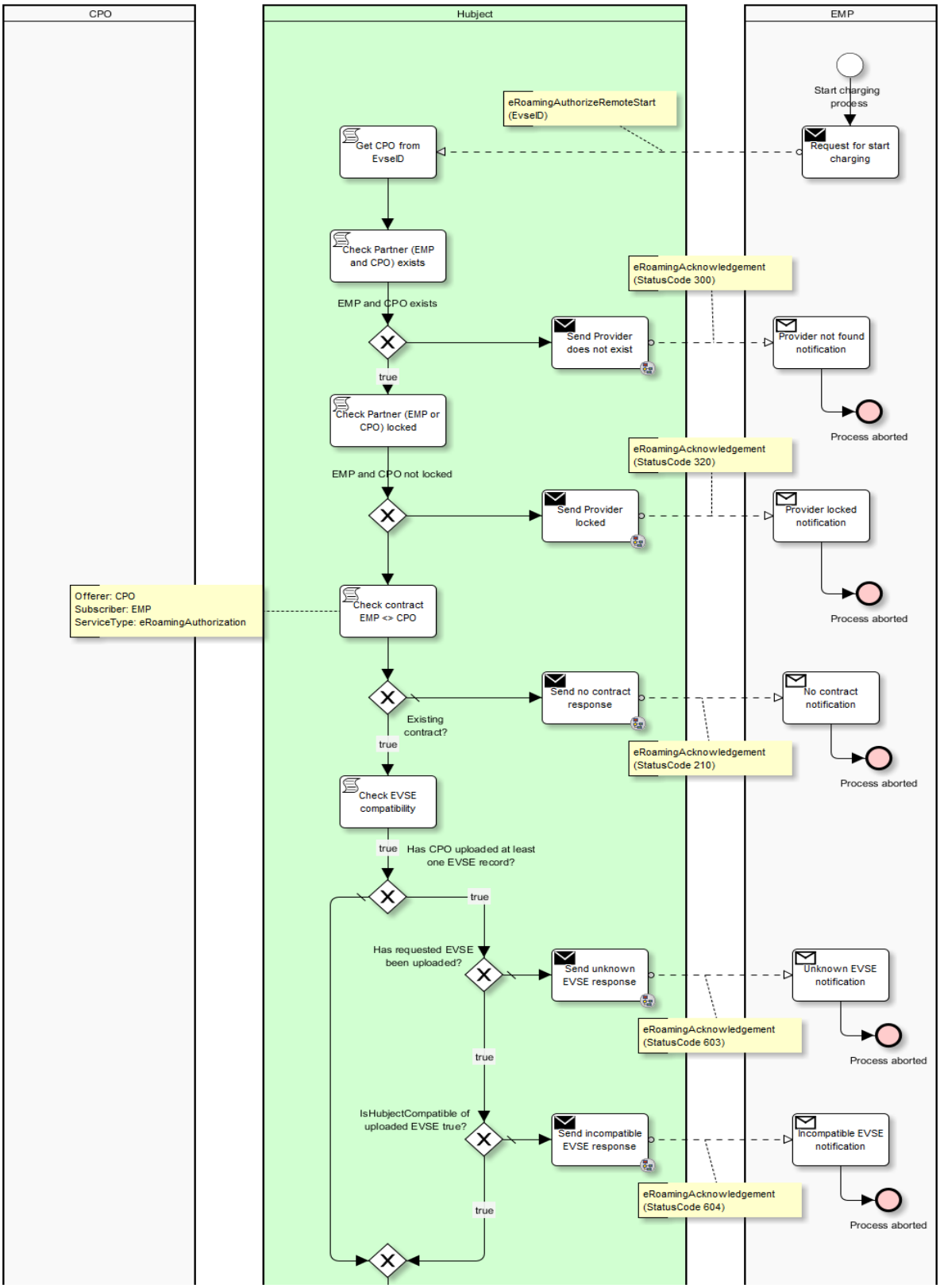


Appendix

Business Process Diagram eRoamingAuthorization

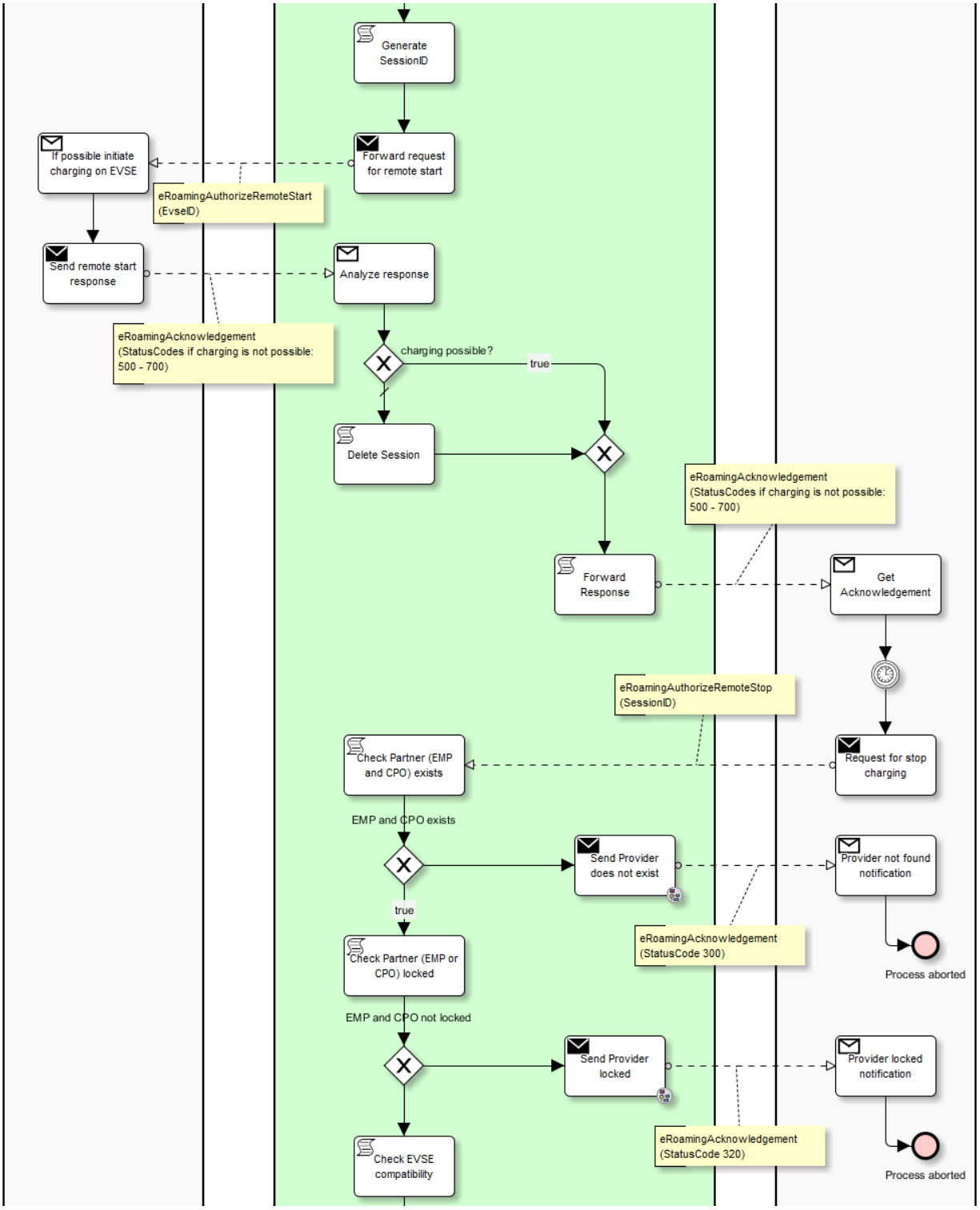


6.2 Business Process Diagram eRoamingAuthorization Remote



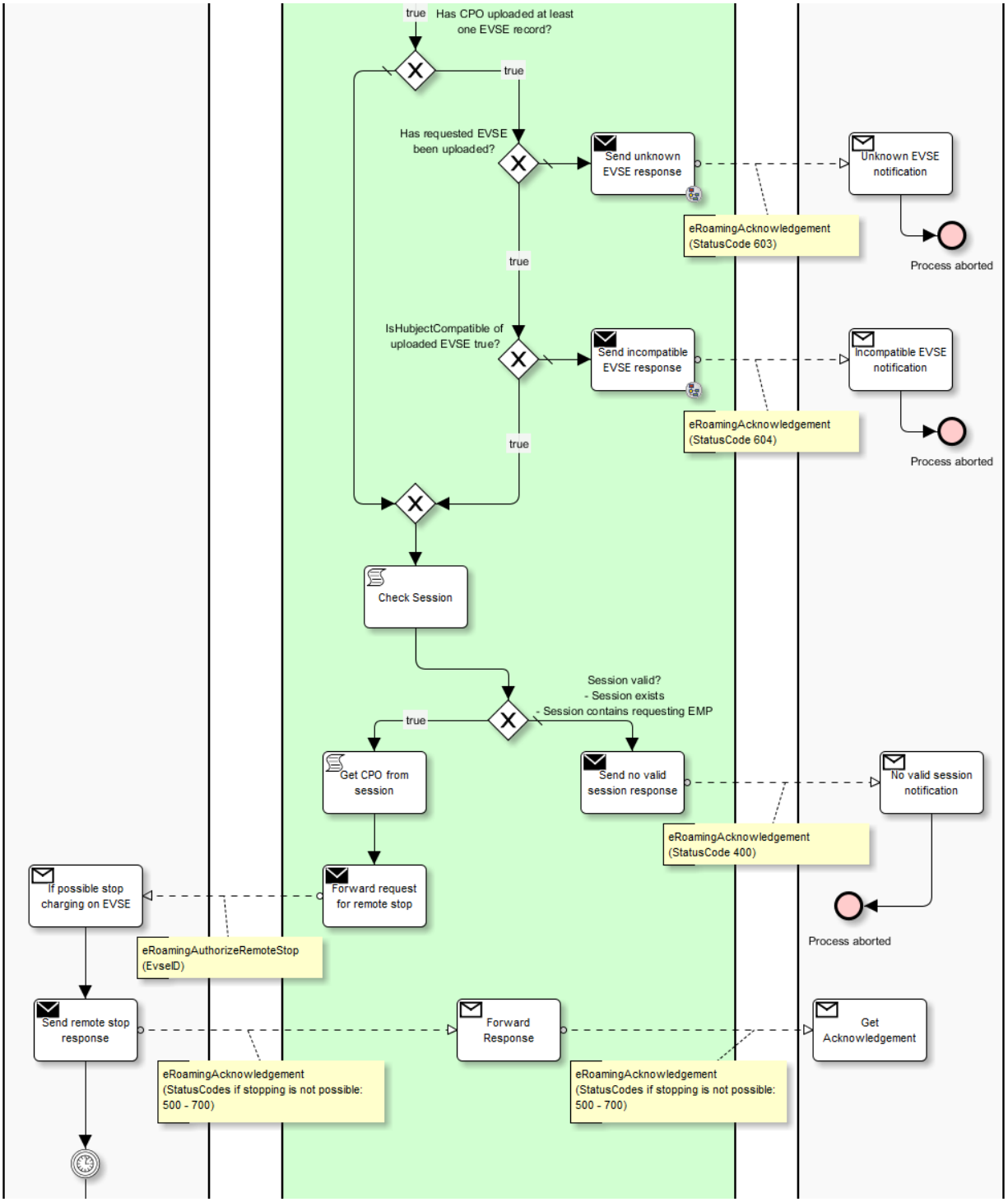
Appendix

Business Process Diagram eRoamingAuthorization Remote



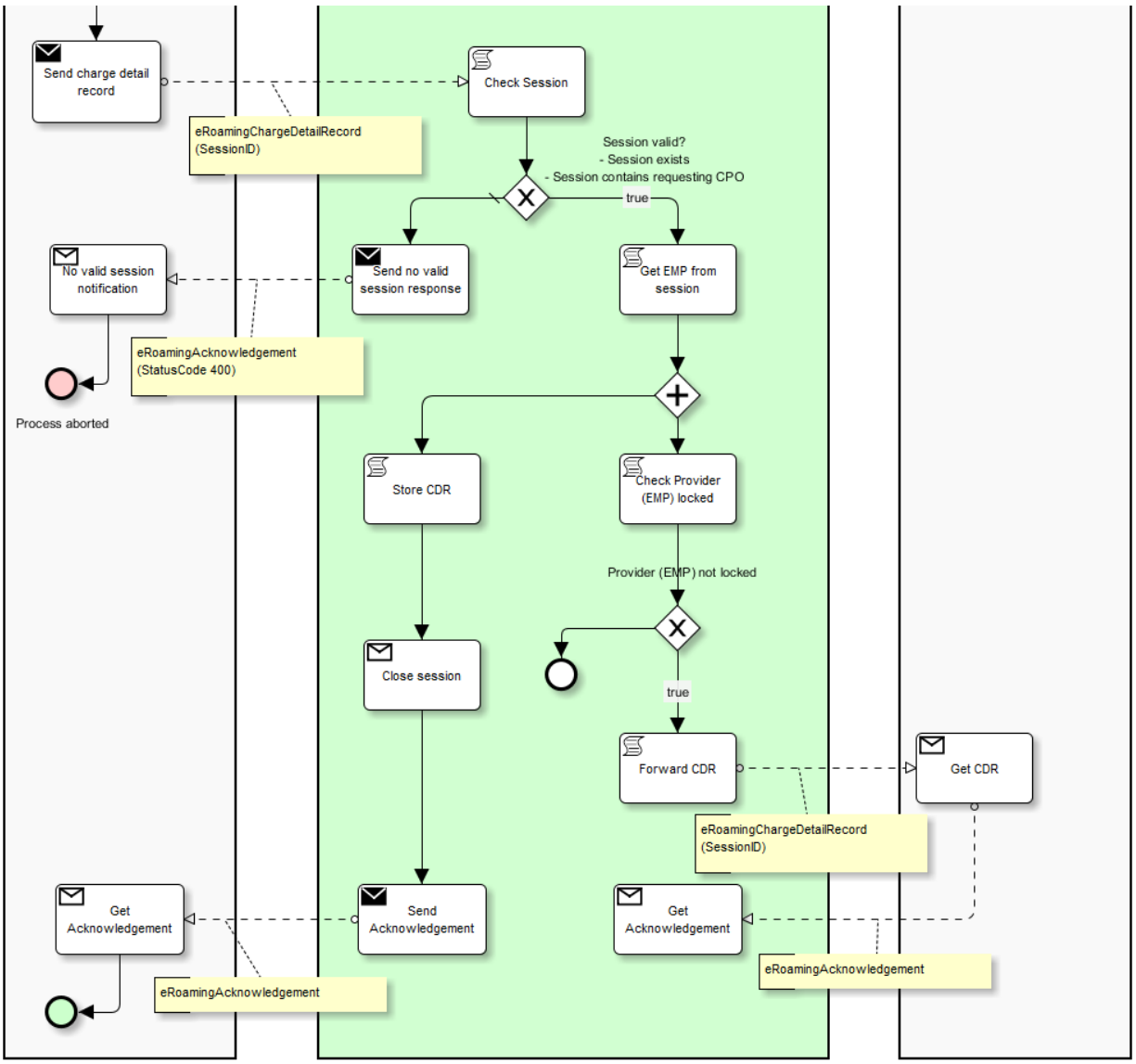
Appendix

Business Process Diagram eRoamingAuthorization Remote

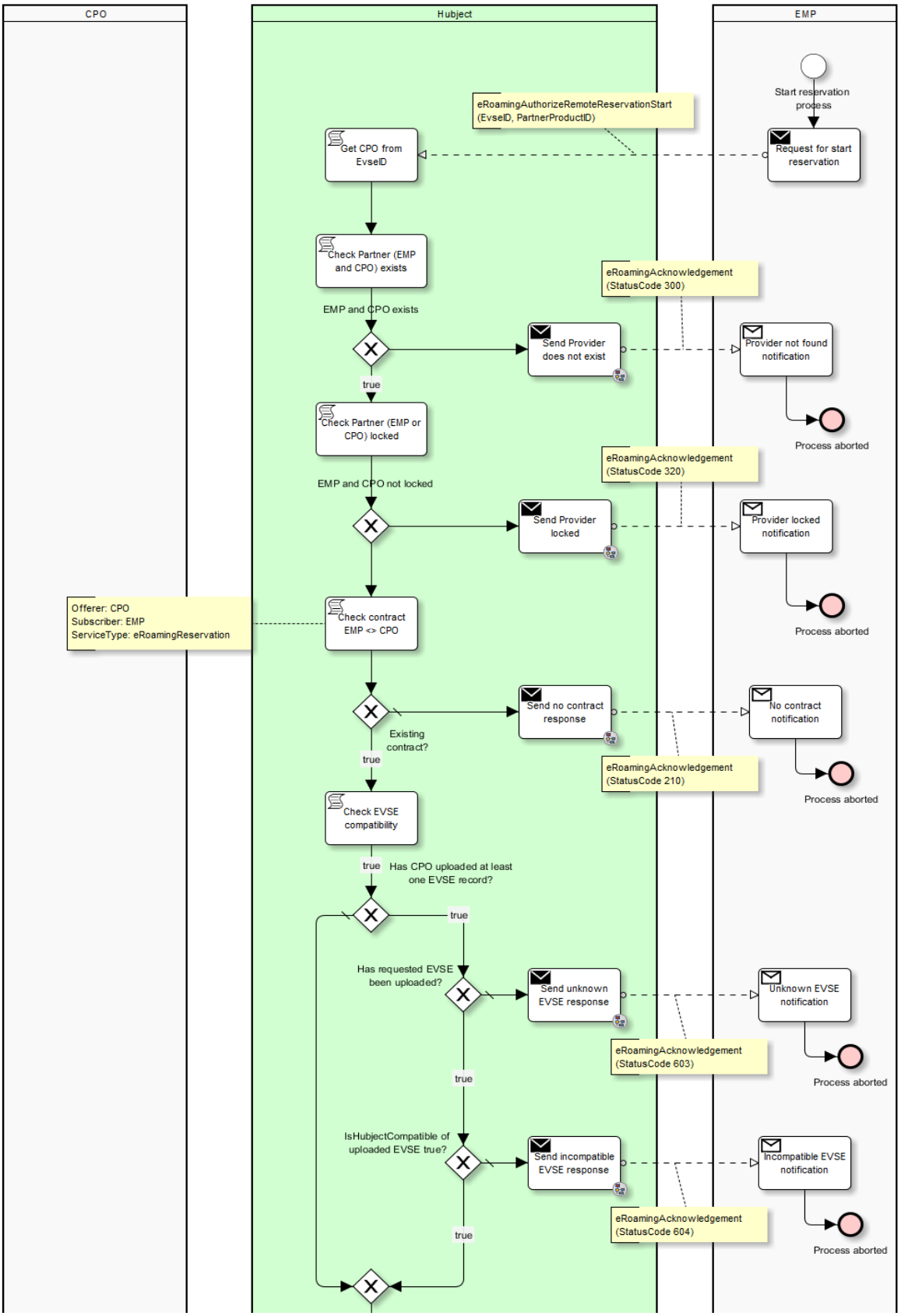


Appendix

Business Process Diagram eRoamingAuthorization Remote

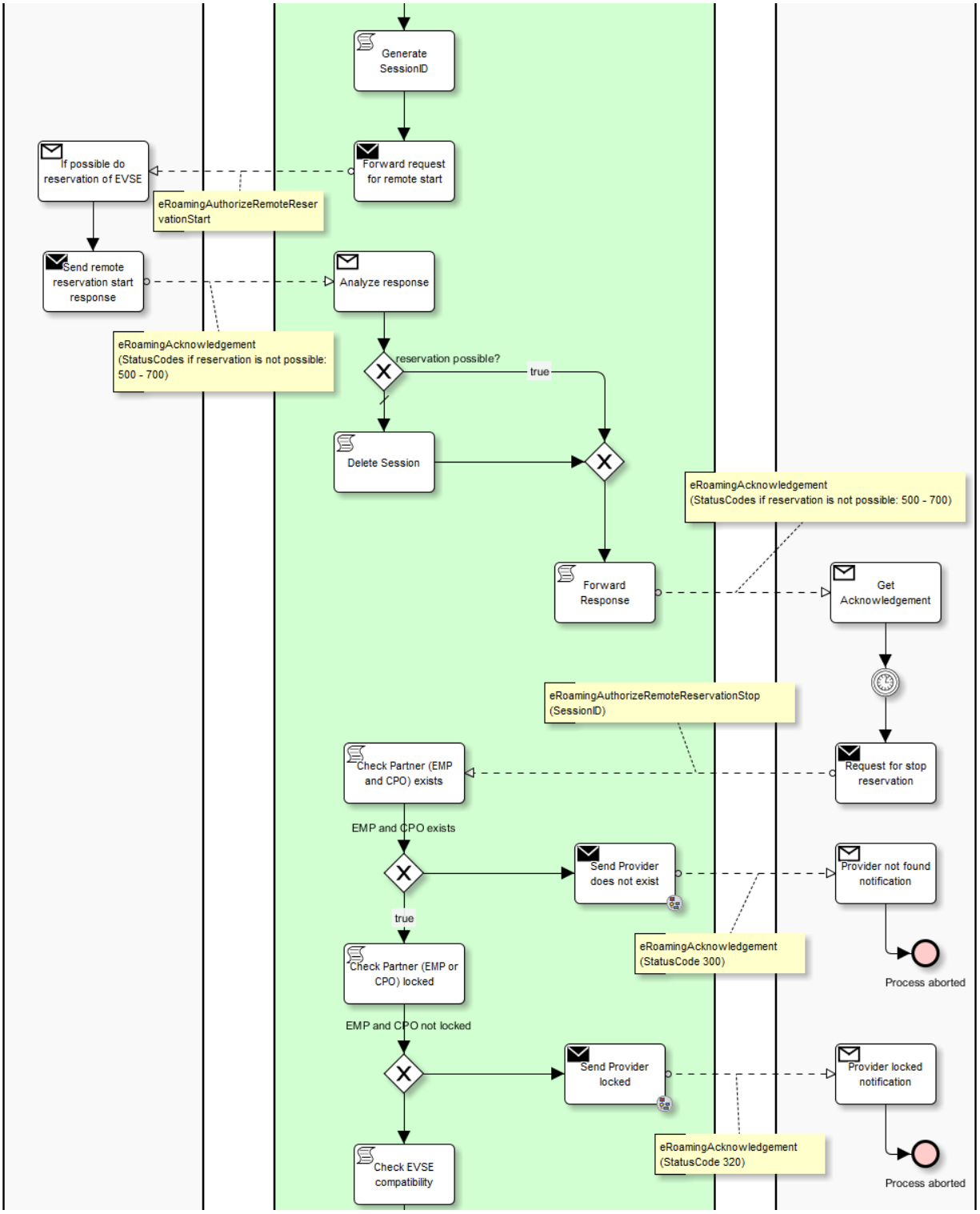


6.3 Business Process Diagram eRoamingReservation



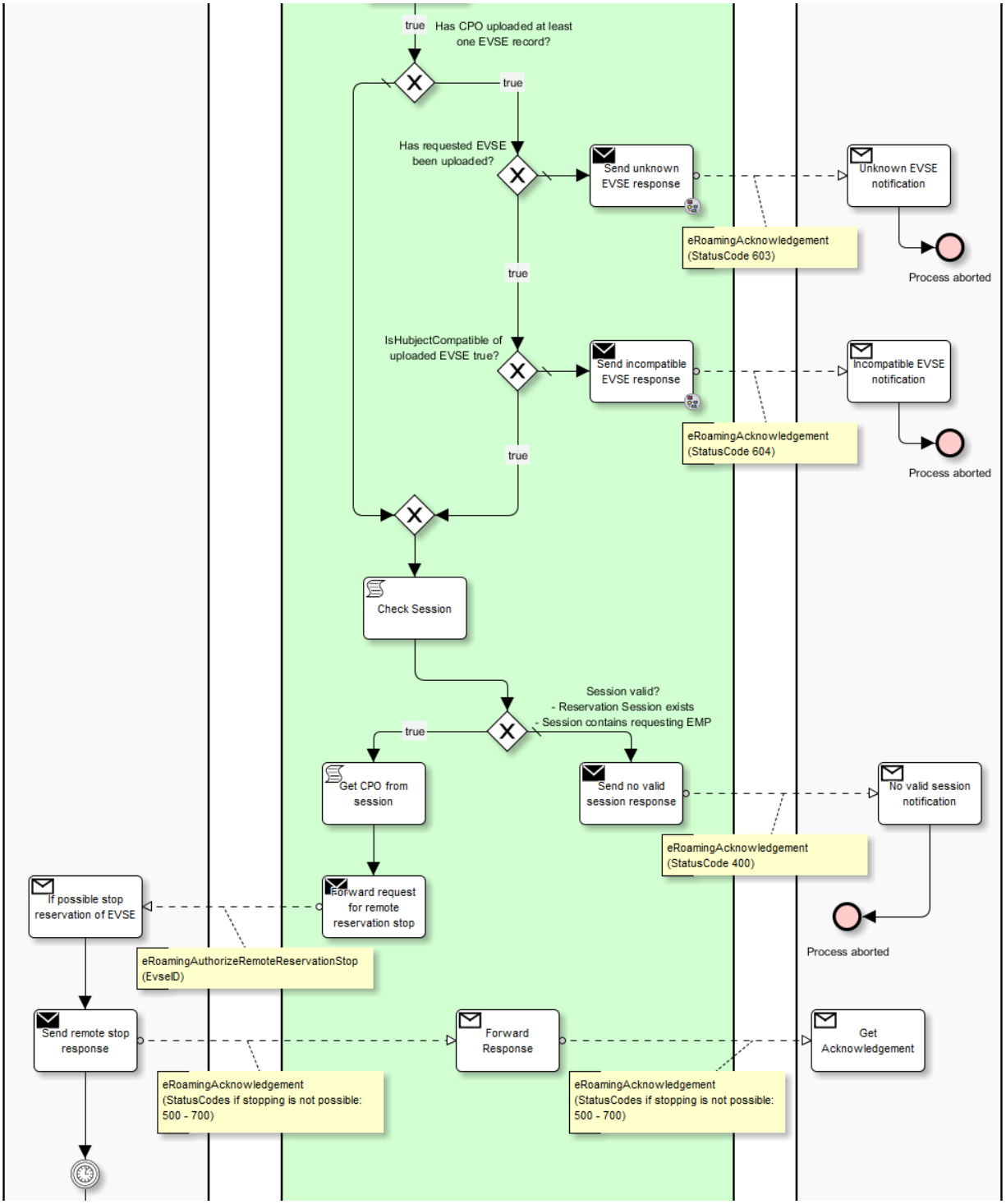
Appendix

Business Process Diagram eRoamingReservation



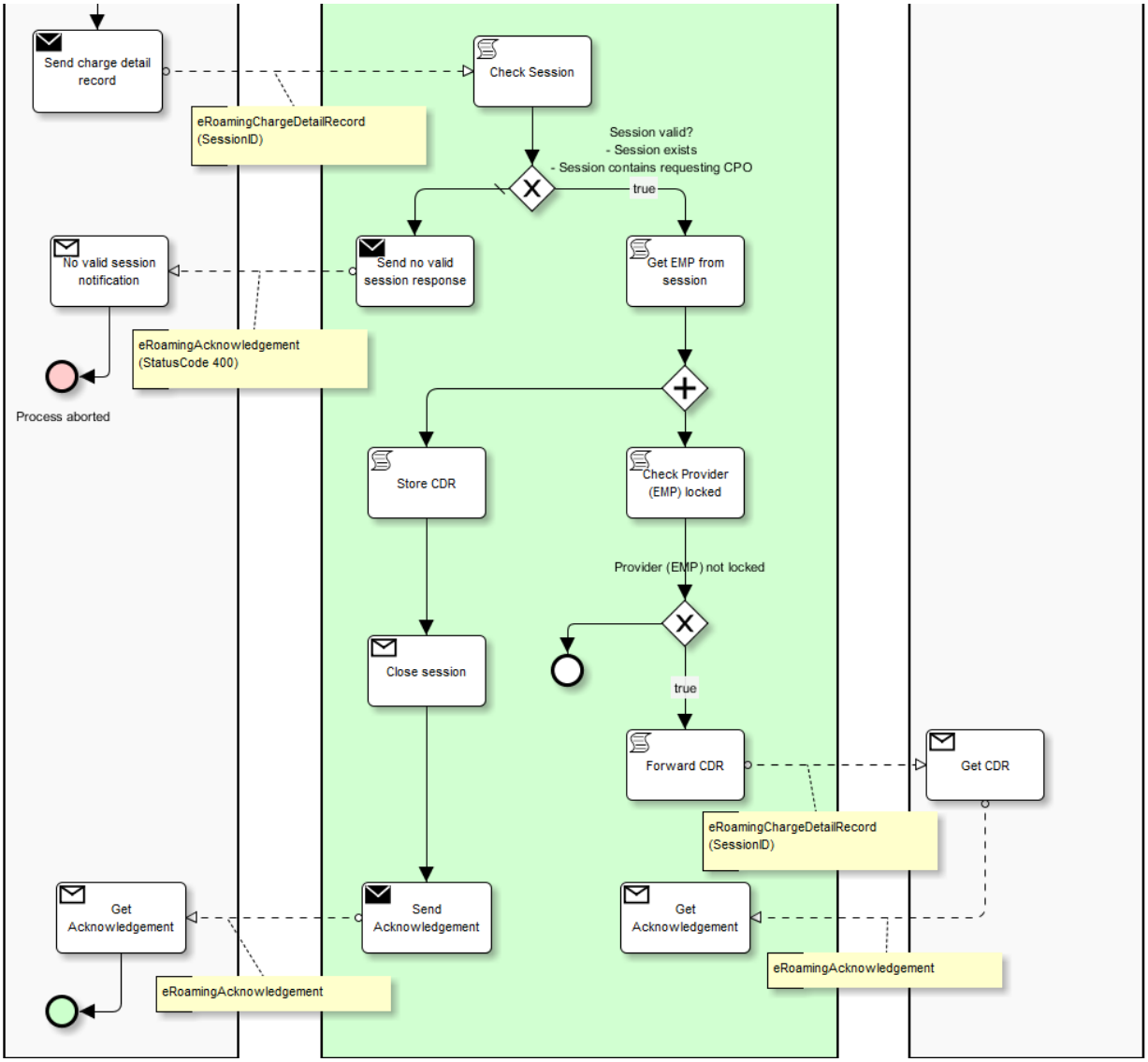
Appendix

Business Process Diagram eRoamingReservation

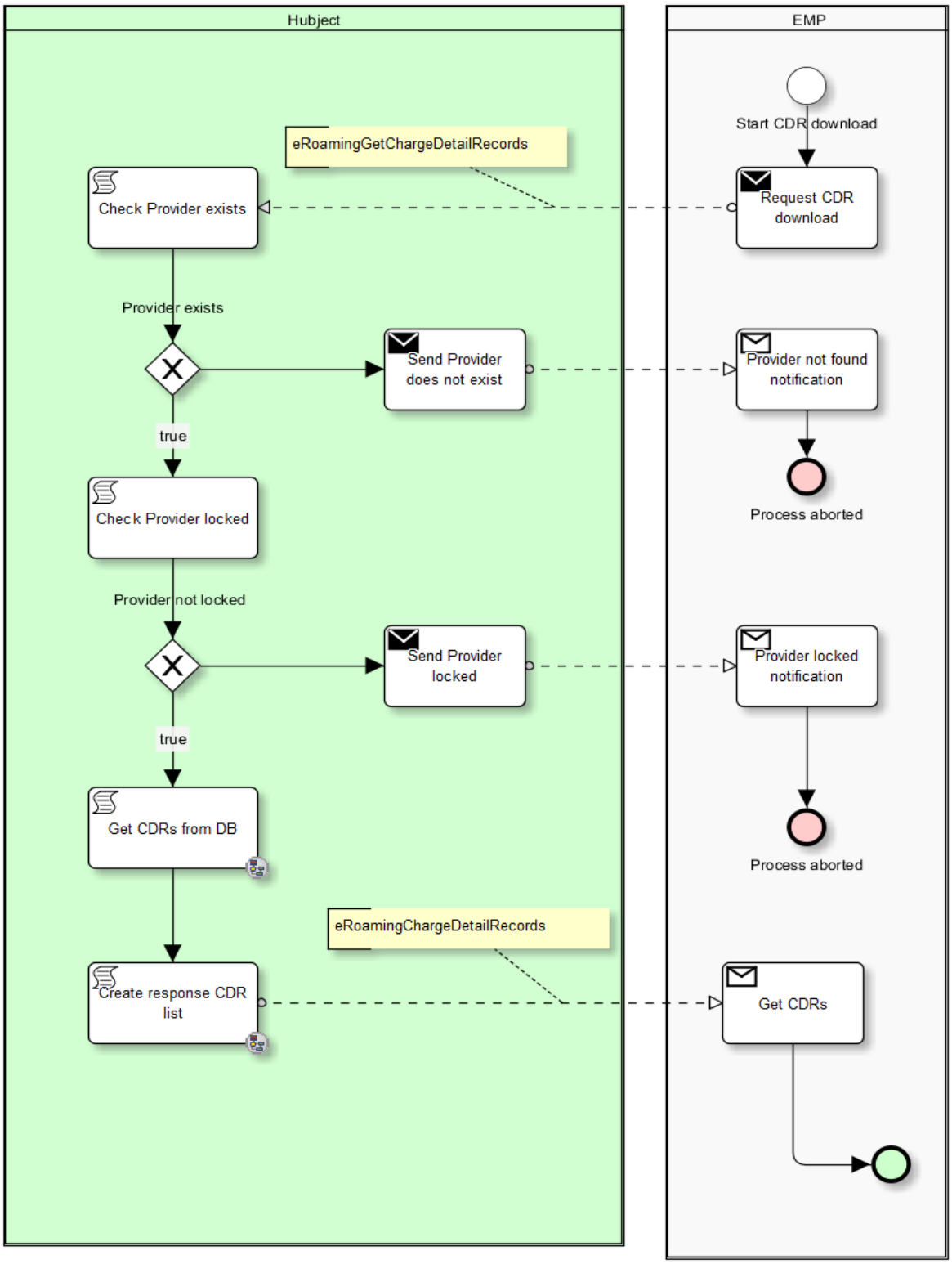


Appendix

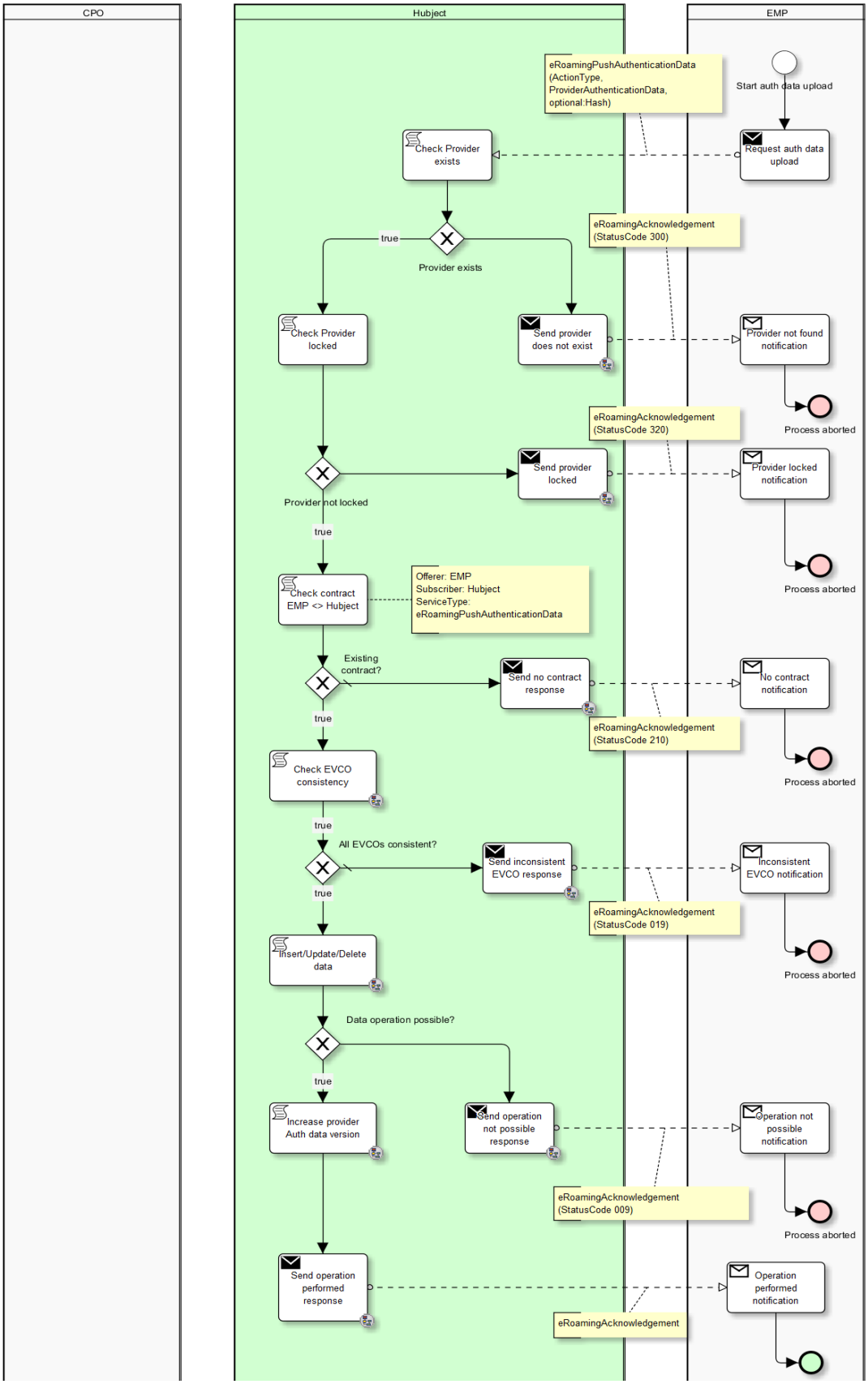
Business Process Diagram eRoamingReservation



6.4 Business Process Diagram eRoamingAuthorization GetCDRs

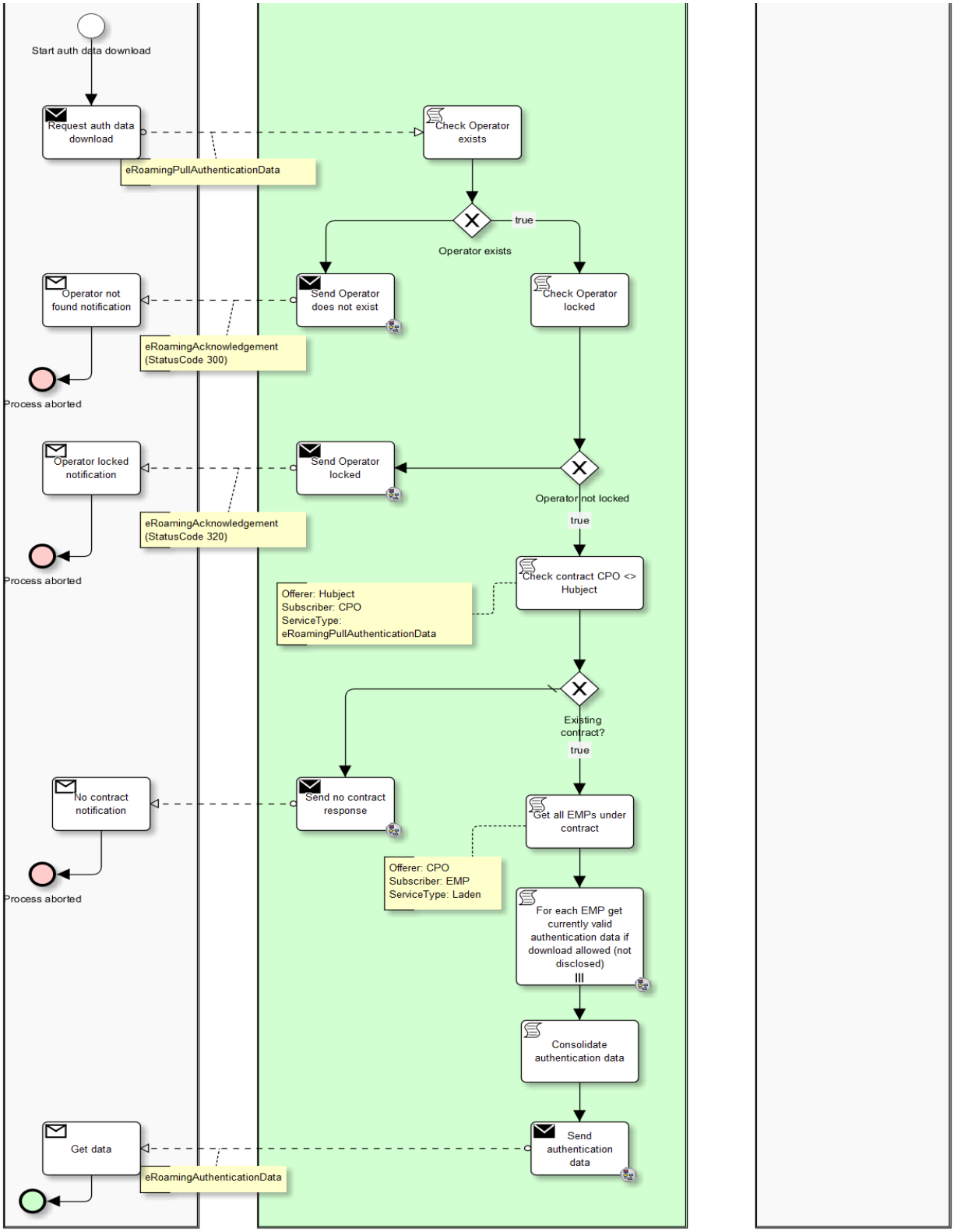


6.5 Business Process Diagram eRoamingAuthenticationData

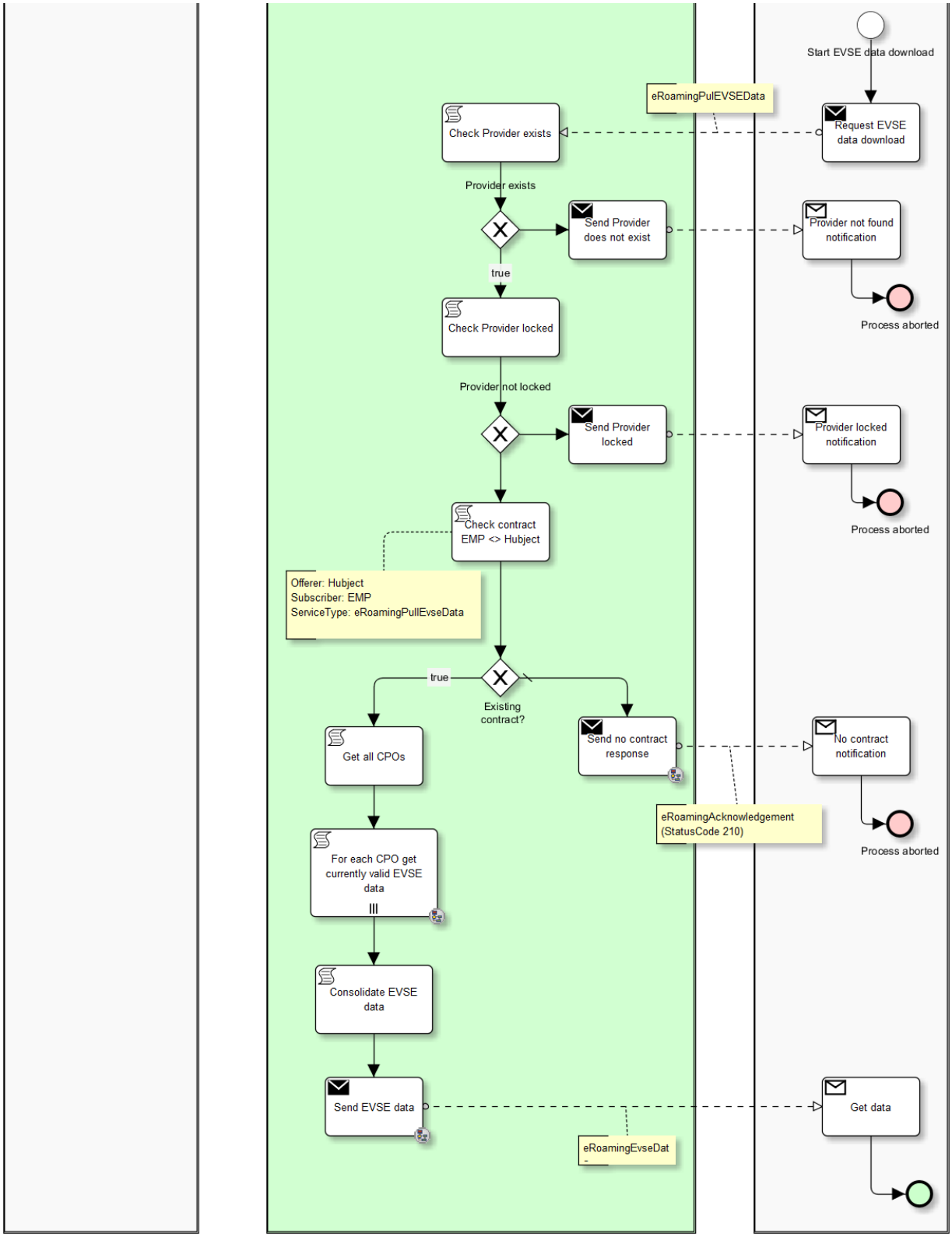


Appendix

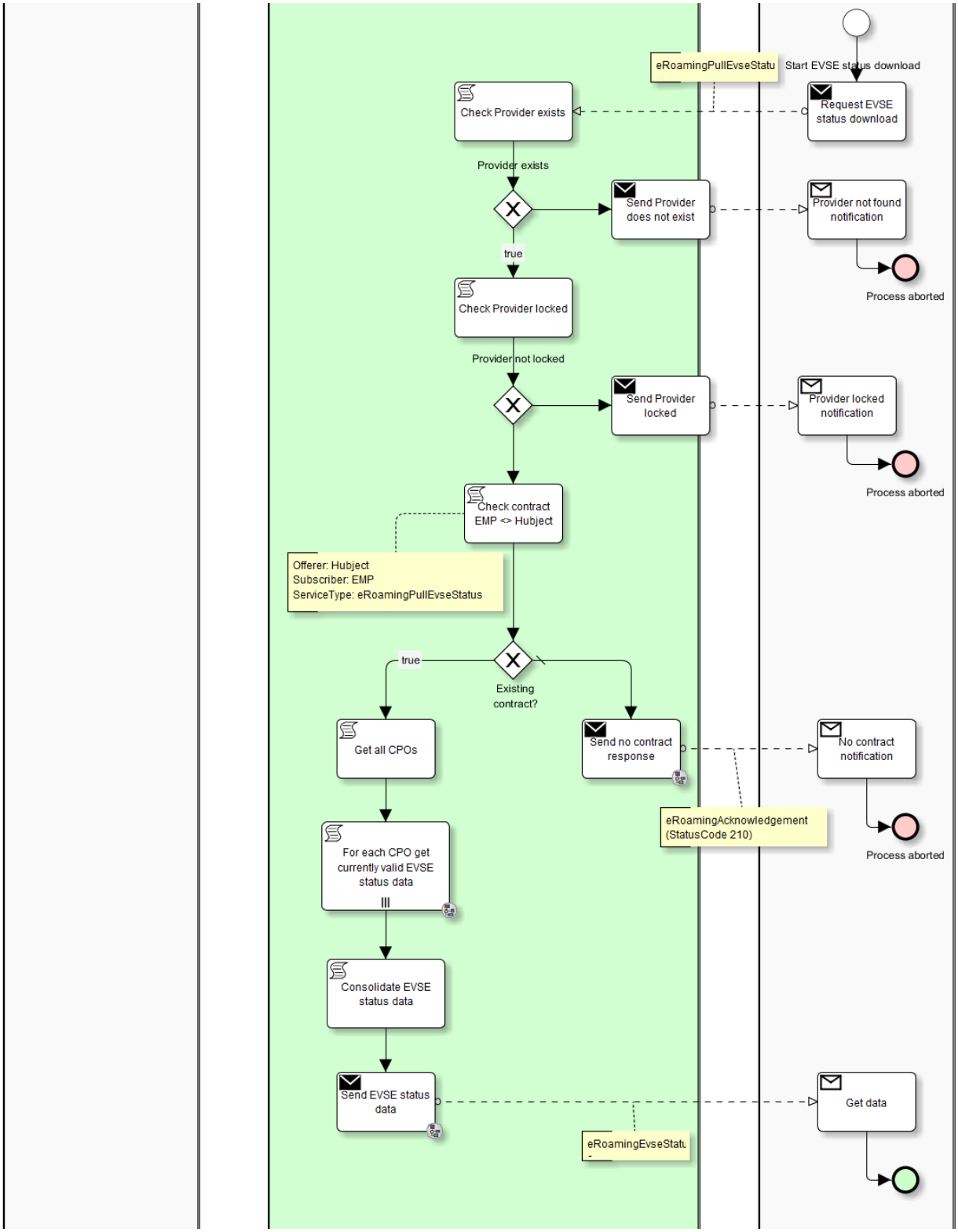
Business Process Diagram eRoamingAuthenticationData



6.6 Business Process Diagram eRoamingEVSEData

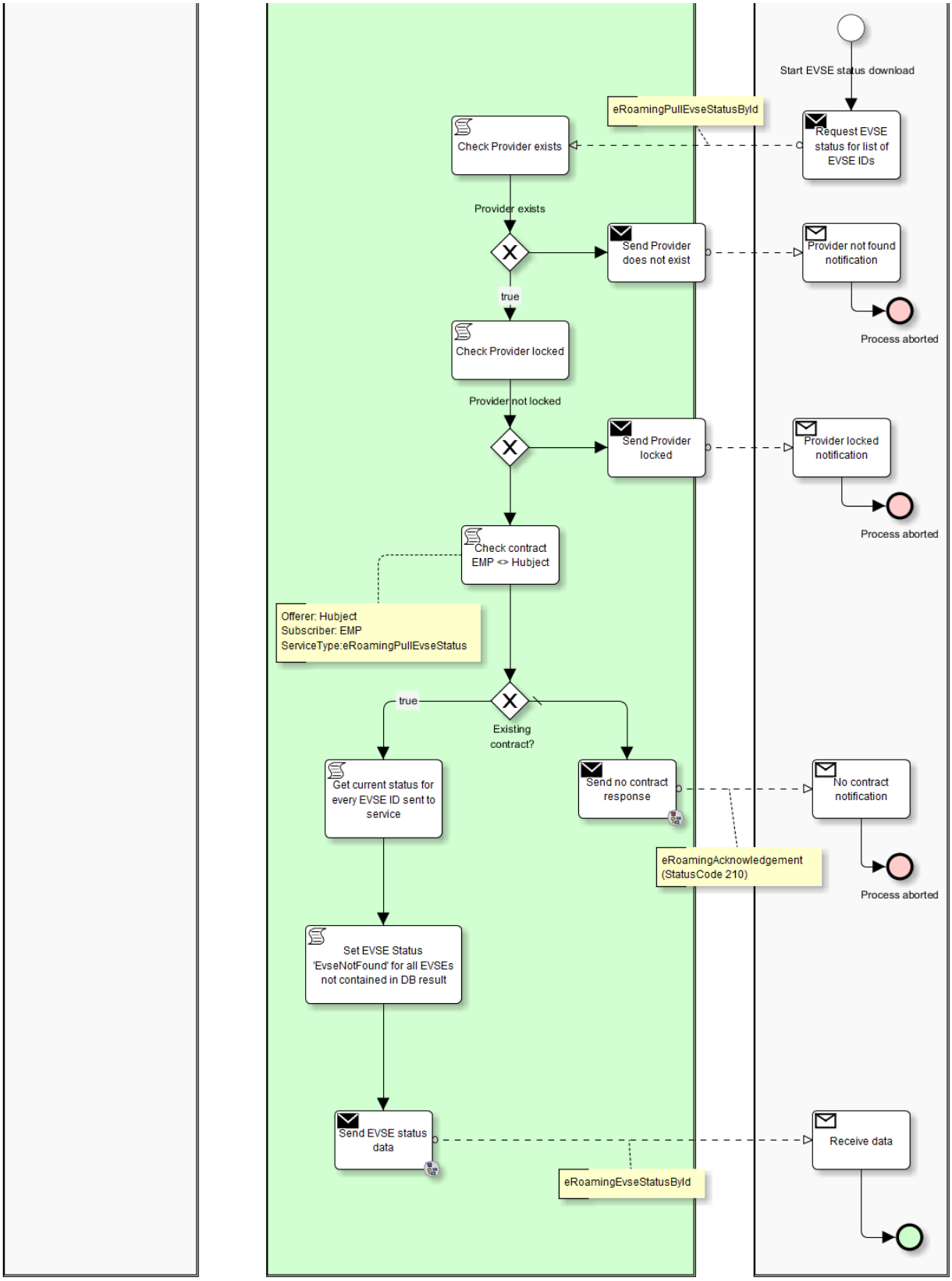


6.7 Business Process Diagram eRoamingEVSEStatus



Appendix

Business Process Diagram eRoamingEVSEStatus



7 Glossary and Abbreviations

Charging Station	The unit where an electric vehicle is charged. A charging station consists of one or more charging spots (EVSE).
CPO (Operator)	Charge Point Operator: Mobility partner who operates the charging infrastructure.
EMP (Provider)	Electric Mobility (emobility) Provider: Mobility partner who provides emobility services to customers.
EVCO	Electric Vehicle Contract: Contract between an EMP and a customer.
EvcoID	Electric Vehicle Contract Identifier.
EVSE	Electric Vehicle Supply Equipment: Charging spot.
EvseID	Electric Vehicle Supply Equipment Identifier.
GUI	Graphical User Interface.
GUID	Globally Unique Identifier.
Hash / Hash Code	String with a fixed length that represents a data set. The hash code is generated by applying a hash function (e.g. SHA-1 hash function) to the original data.
Hubject Brokerage System (HBS)	The Hubject B2B system is the central software component that routes or storesservice information between mobility partners.
Marketplace	The role “Marketplace” is bound to the central the administrative function of the HBS system.
Mobility partner system	A mobility partner system is the central software component of a Mobility Service Provider (EMP or CPO) and operates e.g. the charging infrastructure or the electric vehicles of the Service Provider.
Session	Web service operations can be bundled and related to a certain session by unique IDs.
SHA-1	Secure hash algorithm: A cryptographic hash function that is used to map data values to fixed-length key values.

Glossary and Abbreviations

SOAP	Simple Object Access Protocol: A web service standard that specifies the implementation and information exchange of web services.
SSL	Secure Socket Layer:
UTF-8	A protocol for encrypting information over the Internet.
WGS 84	World Geodetic System (1984): A standard coordinate frame which is used to represent geo coordinates used by the GPS system as reference coordinate system.
WSDL	Web Service Definition Language: Technical description of functionality that is offered by a web service.
XML	Extensible Markup Language: A technical language that defines the format and encoding of documents for data exchange.