Open InterCharge Protocol
for
Emobility Service Provider

Version 2.1

May 2016

Table of Contents

Table of Contents

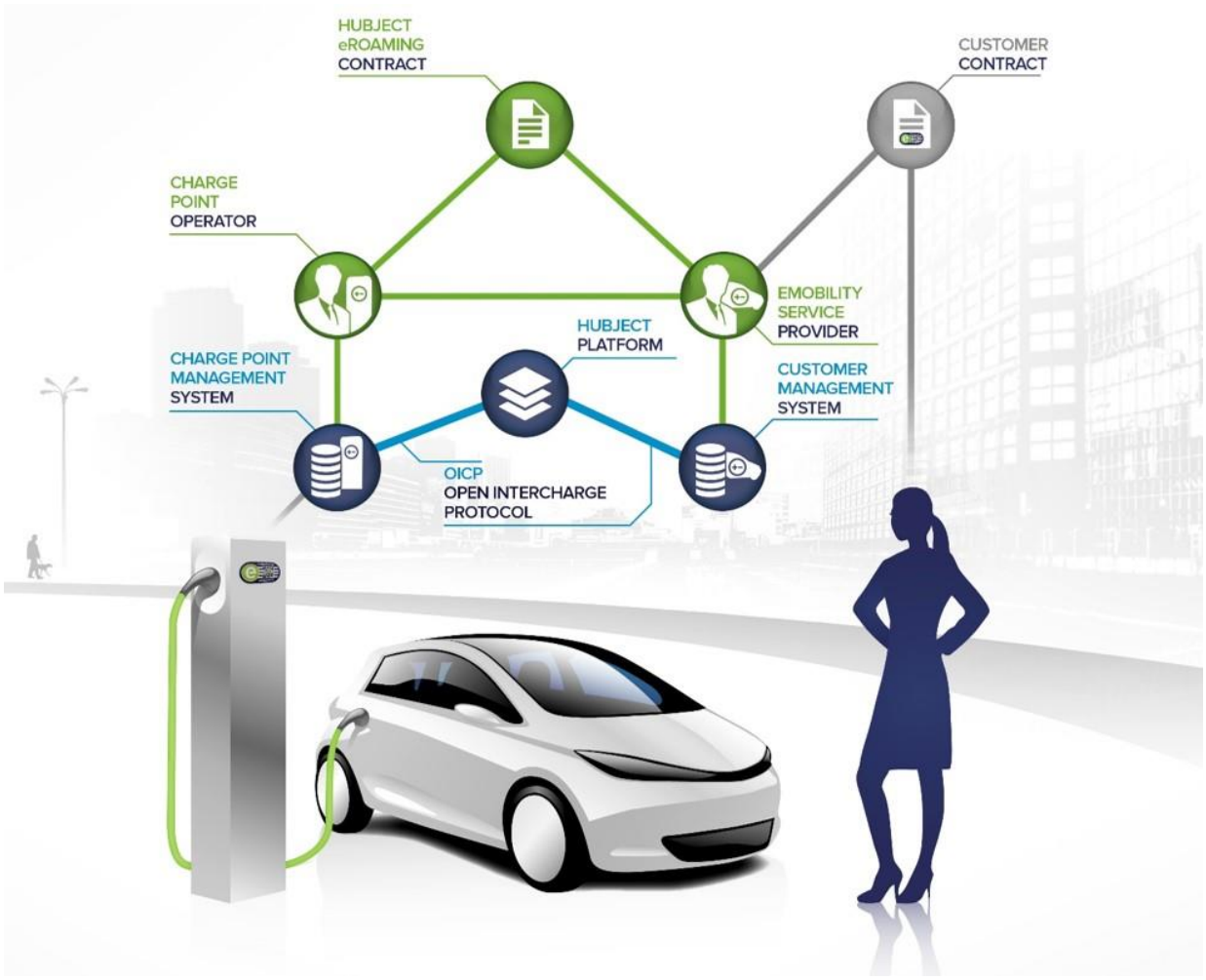
Table of Contents

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 in Europe 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 at the Hubject platform via the eRoamingPushAuthenticationData (see 4.1.1). The HBS will authorize charging sessions locally at the platform. Nevertheless there can be a realtime connection for pulling dynamic POI data.4.1.3). Furthermore, the CDRs, as a result of each charging session, will not be directly forwarded to the EMP but will be stored also at the Hubject system. The EMP than can download this CDRs in line with his demands via the eRoamingGetChargeDetailRecords (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 will be forwarded to the EMPs system. To ensure the online authorization the EMP MUST implement the holistic authorization web service (see 3.1). Furthermore, the CDRs, as a result of each charging session, will be directly forwarded to the EMPs system. Therefore, the CPO MUST implement the eRoamingChargeDetailRecord
(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 European 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 that there is a valid contract for the requested service. The way how EMPs and CPOs manage their service contracts is out of scope of this document because contract management is conducted via a GUI-based system component.

Introduction

This new OICP Version 2.1 contains new services and data types to enable new business processes for the Partners who uses the HBS all over Europe.

1.4 Conventions

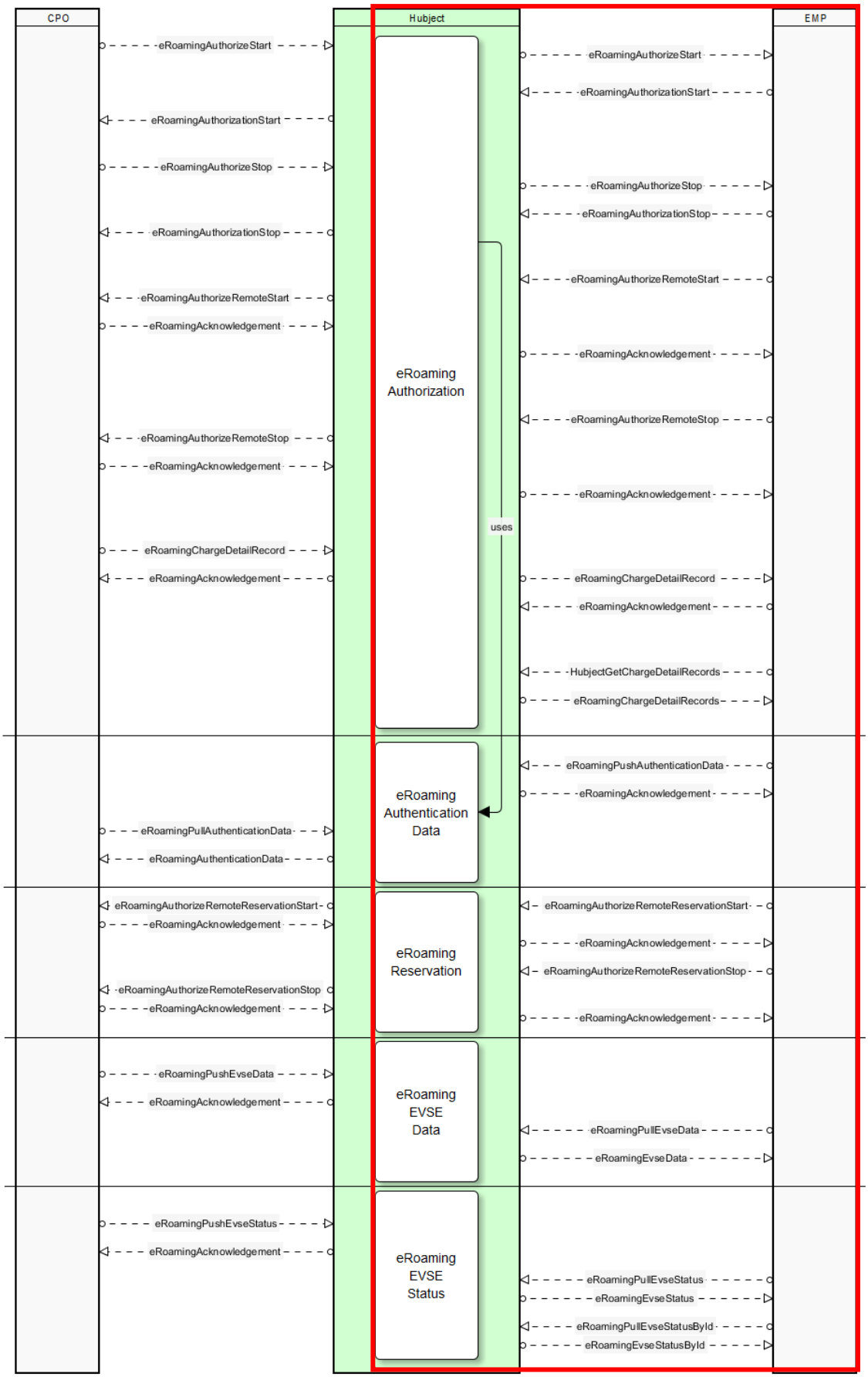
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119].

1.5 Overview

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

Chapter 3 introduces the supported web services and the corresponding service operations in detail. Chapter 4 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

The release of the new OICP version utilizes the newly included concept of the HBS-Release management. This allows the reduction of the number of interface changes and makes them more predictable. Therefore spontaneous Releases can be avoided. This means, that two times a year a new version of the HBS is released:

- April the 1st of each year there MAY be a new release of the HBS without interface adaptation
- October the 1st of each year, there MUST be a new release of the HBS, included changing the interface description

The validity of a new OICP Version MUST be supported for two years. This means the newly introduced OICP Version 2.1 can be used from October the 1st, 2016 and is available for usage till September the 30th, 2018. At October the 1st, 2017 there MUST be a new release of the HBS, which is available till September the 30th, 2019.

Each Partner who uses the platform MUST deposit the OICP version used by him in his system and which is used to communicate with HBS. At the database level, the current guidelines are to be implemented in the system, i.e. new mandatory fields should be filled with a value. To communicate between the versions, the following data matching scenarios are used and are mapped by the HBS.

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.

- **"Release notes for the OICP Version 2.1"** - Contains all interface related user stories, which were developed for OICP Version 2.1, all not-interface related stories are not shown.
- **"Mapping Documentation"** - Contains the description of all mappings which are necessary due to interface modifications from OICP-2.0 to OICP-2.1
- **"The Hubject Wiki" –** Contains all relevant information regarding the onboarding process and the technical information holistically

1.8 OICP protocol version and service versions

Until OICP version 2.0 the overall functional OICP protocol version and the technical web service versions, which are part of the service endpoints and the namespaces, have been the same. So OICP 2.0 only contained services with version 2.0.

With the OICP protocol version 2.1 and the introduction of the new reservation service this version correlation has been abolished. This allows more flexibility and avoids unnecessary changes. Thereby, it minimizes implementation work at partner side, in case that a service does not change functionally with a new OICP version.

The table shows all web services and their current version as part of OICP version 2.1.

Service	Version
eRoamingAuthorization	2.0
eRoamingReservation	1.0
eRoamingAuthenticationData	2.0
eRoamingEvseData	2.1
eRoamingEvseStatus	2.0

2 Communication paradigms

2.1 SOAP

The service communication between provider systems and Hubject is based on the most widely used web service standard SOAP 1.1. Due to this, the technical interface description is represented by WSDL files that are created for every eRoaming service, which supplement this document.

All described web services 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 will be validated against the corresponding WSDL (technical service definition). In case that a request does not match the WSDL, Hubject responds with a standard SOAP fault message indicating the format violation.

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.

Communication paradigms

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 eRoamingAuthorize request fails, the requested provider can e.g. specify why the user cannot be authorized.

Chapter 5.2.11 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).

Communication paradigms

2.6 Session handling

Some web service operations that are defined in chapter 3 together form a functional business process, respectively a functional session.

Example:
The operations in eRoamingAuthorization (see 3.1) cover a charging session. A charging session can be started with eRoamingAuthorizeStart or eRoamingAuthorizeRemoteStart operations and stopped with the corresponding operations. Consumption data belonging to a charging session can be sent using the eRoamingChargeDetailRecord operation.

In order to be able to relate operations to the correct session, Hubject assigns a SessionID to every session after the reception 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 has not been created by Hubject or that is not valid, the request receives negative response and no further process steps will be conducted.

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 a second (optional) request parameter PartnerSessionID. CPO partner systems can use this parameter to send their own session IDs. Hubject will assign a PartnerSessionID to the Hubject SessionID and will add the PartnerSessionID to every operation response so that the CPO partner systems can relate the operations to their own session handling.

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-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.

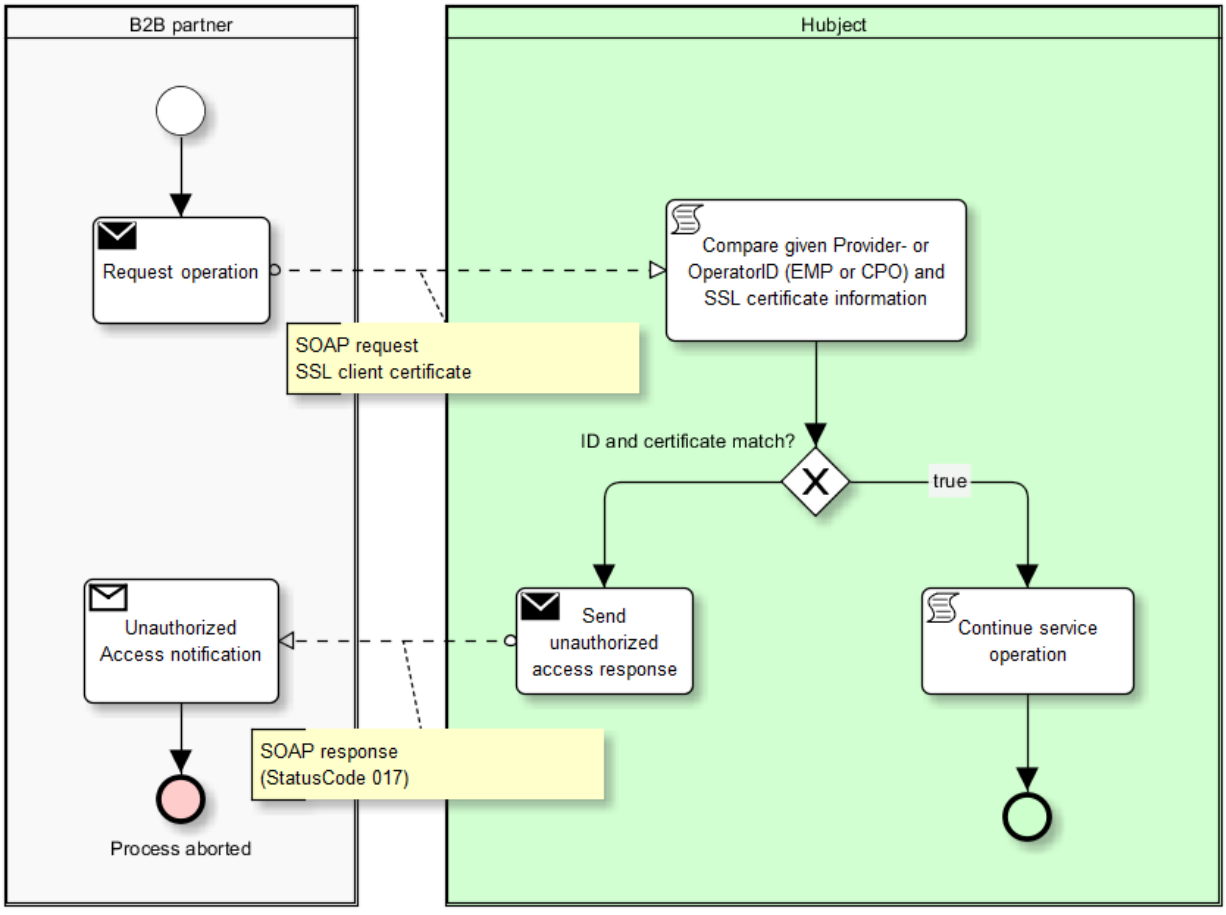
The IDs are assigned to a specific partner role and they are cross-market unique. There are two roles: eMobility provider (EMP) and charge point operator (CPO). A partner can have one or both roles. In case that a partner has both roles, two IDs (ProviderID see 5.3.3 and OperatorID see 5.3.4) will be 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.

Communication paradigms

If the appropriate ProviderID and OperatorID can be provided implicitly through EVCO- or EVSEIDs, EVCOIDs contain the corresponding EMP's ProviderID. 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. Thus Hubject makes sure that a partner cannot request operations in the name of another partner by simply sending another partner's ID. In case that Hubject detects a mismatch of ProviderID/OperatorID and the client certificate information used with the request, Hubject will not perform the operation and will respond with the status code 017 "Unauthorized Access".

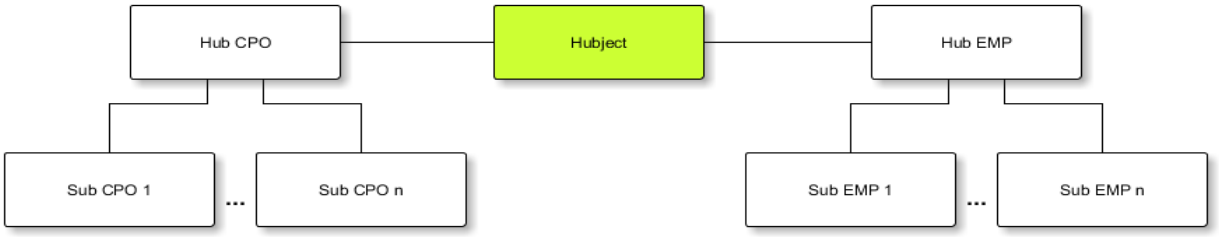
The following process diagram describes the partner identification process. It is performed at the beginning of every web service operation, which is described in this document. Correspondingly the diagram amends all process diagrams in the appendix.



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 order to communicate with Hubject. The following diagram shows the relationships between Hubject, hub partners, and sub partners.

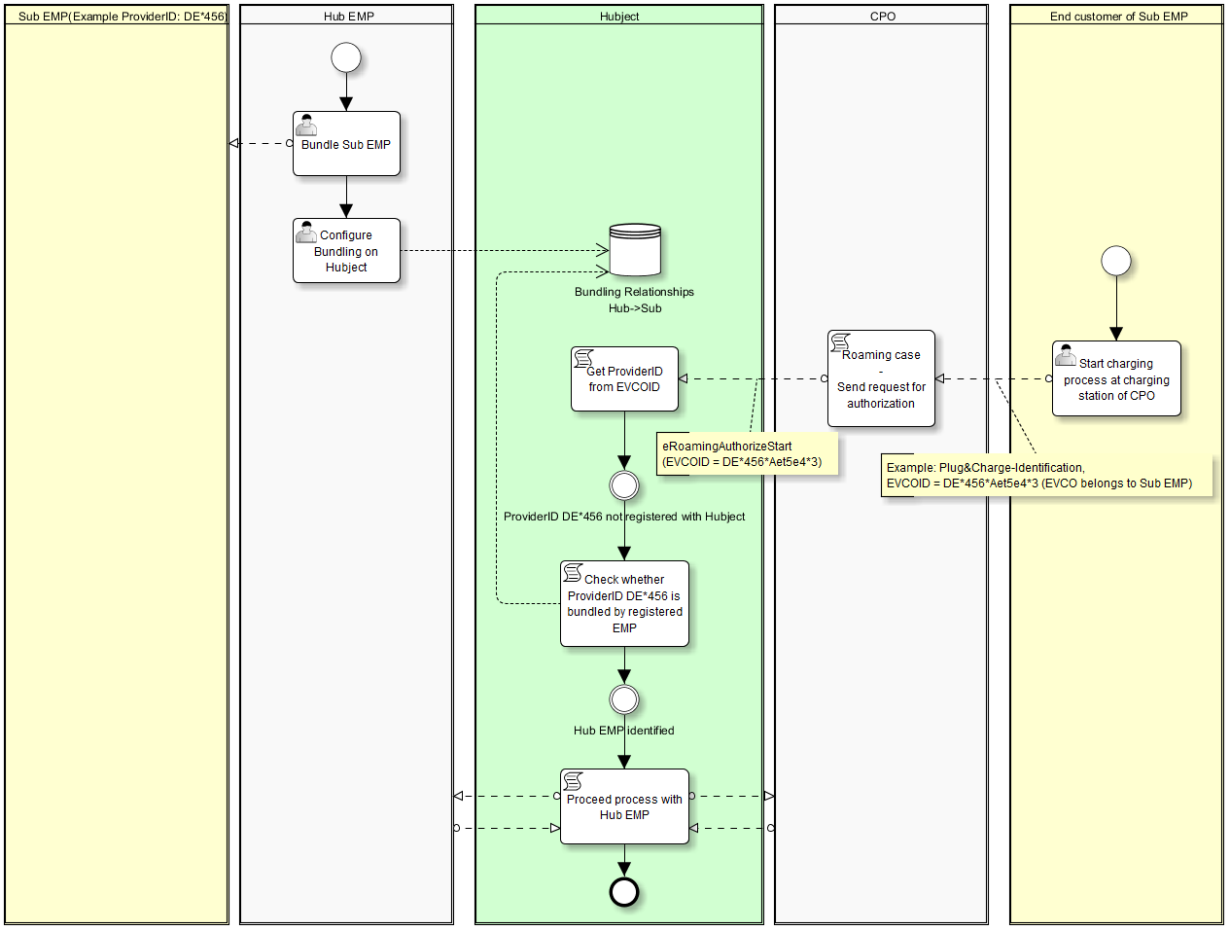


Hubject may receive service requests that contain sub partner information, e.g. an EVCOID containing the ProviderID of a sub partner. In such cases - when Hubject does not find the ID within the group of registered partners – Hubject will check whether the corresponding partner is bundled by a registered 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 2.7 MUST always provide the ID of the actual communication partner. 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.

Communication paradigms



2.9 Data push operations

Hubject offers different operations that allow partners to upload (push) data, e.g. authentication data (see 4.1.1) for the EMP.

In order 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:
Aa EMP sends an AuthenticationData (see 3.3.1) with several hundred data records. Shortly after that, the same EMP sends an Authentication Data full load with only one data record. From the EMP's point of view, the second request should overwrite the first, resulting in only one valid data 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 data records on the Hubject system.

Communication paradigms

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.

In case that Hubject receives messages and stores the data, all date time values will be converted to the Hubject time zone - the time zone were the Hubject system is running. Consequently, all date and time values that can be monitored in the Hubject system implicitly have the Hubject time zone.

In case that partners download data that is stored on the Hubject system (e.g. using the eRoamingGetChargeDetailRecords operation, see 3.1.6), date time field values will always have the Hubject time zone.

3 Services and Operations

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

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

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

Other services will only be consumed by provider system, which means that they do not have to be offered by the provider systems. E.g. the eRoamingAuthenticationData service will only be offered by Hubject.

The names of all following services contain a version and are based on the pattern “<service>_V<version>”, e.g. “eRoamingAuthorization_V2.0”. The reason for this is that Hubject potentially has to offer different versions of a service at the same time. This requires different technical service end points (service URLs). The version in the name guarantees different service end points.

3.1 eRoamingAuthorization_V2.0

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

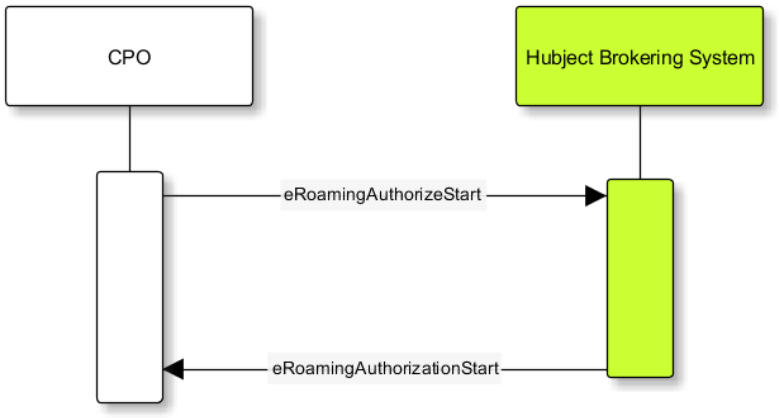
Functional description:

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. Nevertheless, in order to authorize the charging, 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.

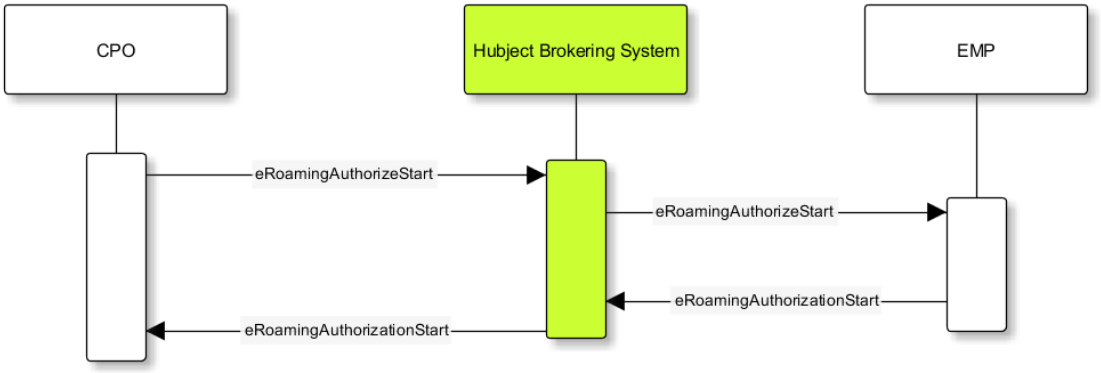
Hubject generates a SessionID for the charging process and persists important session data (SessionID, EVSEID, identification data).

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 eRoamingPushAuthenticationData service (see 3.3.1).

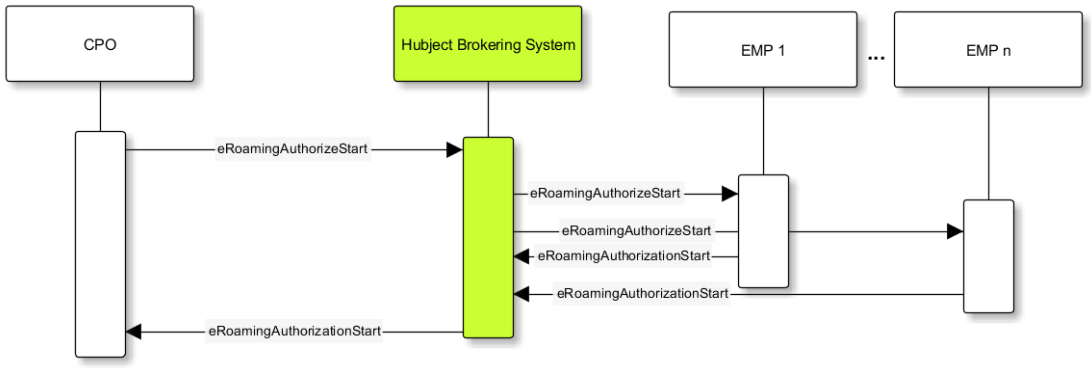


B) In case that offline authorization is not possible, Hubject tries to derive the EMP from the provided identification data. QR Code and Plug&Charge identification data contain the EVCOID. Hubject can derive the EMP's ProviderID from the EVCOID. Hubject will directly forward eRoamingAuthorizeStart requests 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.

Pin Security:

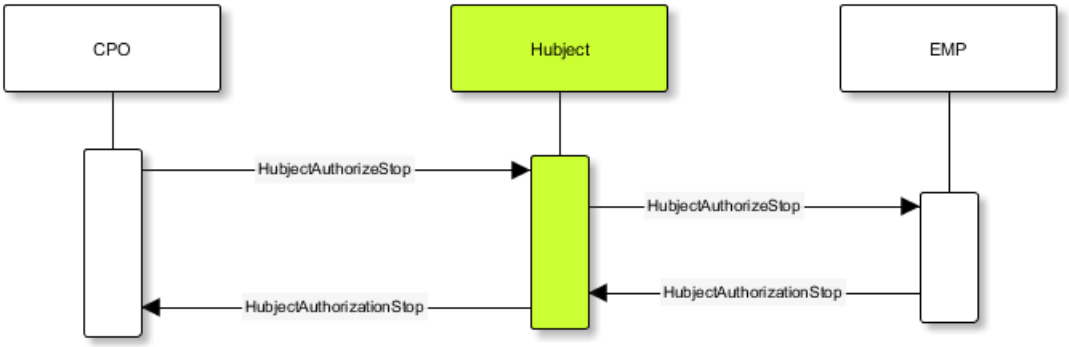
The eRoamingAuthorizeStart request contains one of the defined identification types (see 5.1.2). The identification type “QRCodeIdentificationType” (see 5.1.11) contains - besides the “EVCOID” field - a “PIN” field or a “HashedPIN” field (only one of the two options must be provided).

For security reasons and as a general rule, Hubject does not store PINs in clear text, but always as encrypted hash values (see 3.3.1). 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



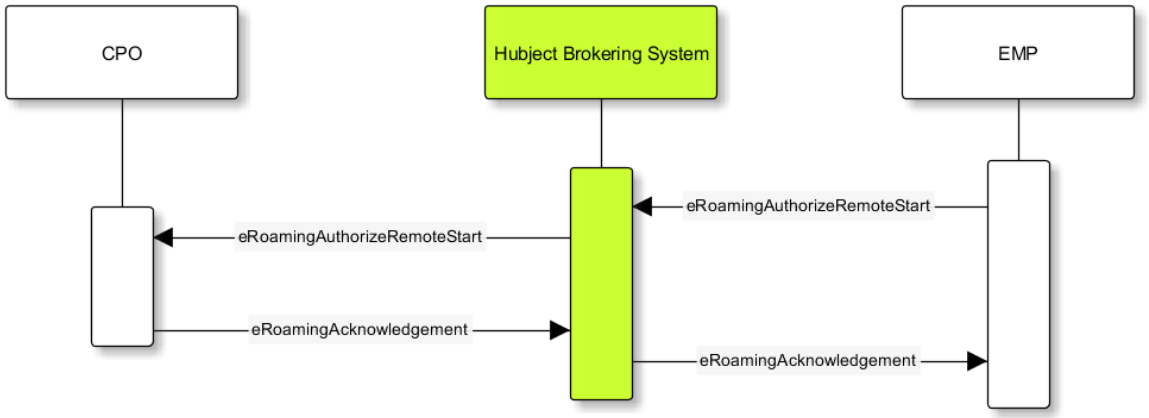
eRoamingAuthorizeStop basically works similar to the service eRoamingAuthorizeStart. The request is sent in order to authorize the stopping of a charging process. The request MUST contain the SessionID that was created by Hubject after the initial eRoamingAuthorizeStart request. In most cases, Hubject can derive the EMP that has authorized the charging from the session and can directly and offline authorize the request or forward the request for stopping to the EMP. In case that the charging session originally was 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



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 a eRoamingAuthorizeRemoteStart request to Hubject. The request MUST contain the ProviderID and the EVSEID.

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

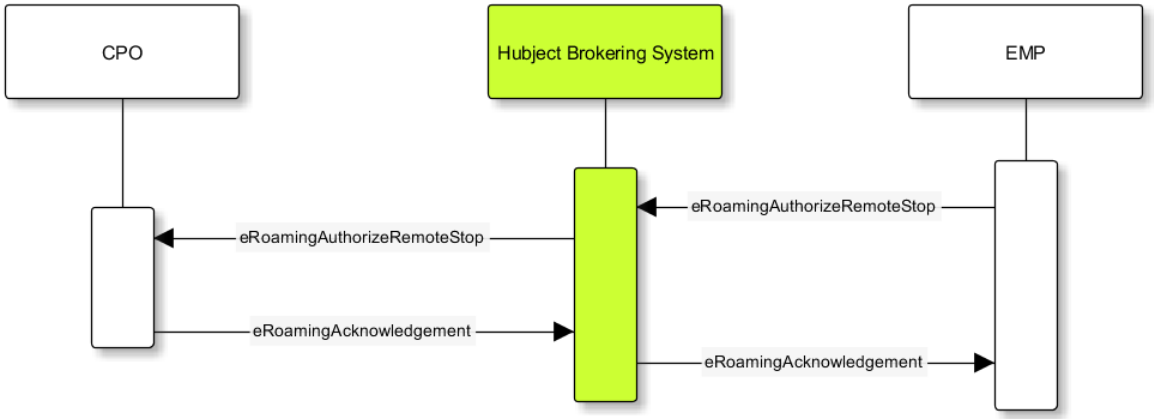
Hubject will check whether there is a valid contract between the two partners for the service (EMP must be the subscriber). If so, Hubject continues with checking the charging spot's compatibility. In case that the CPO has uploaded at least one charging spot data record, Hubject will check whether the 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

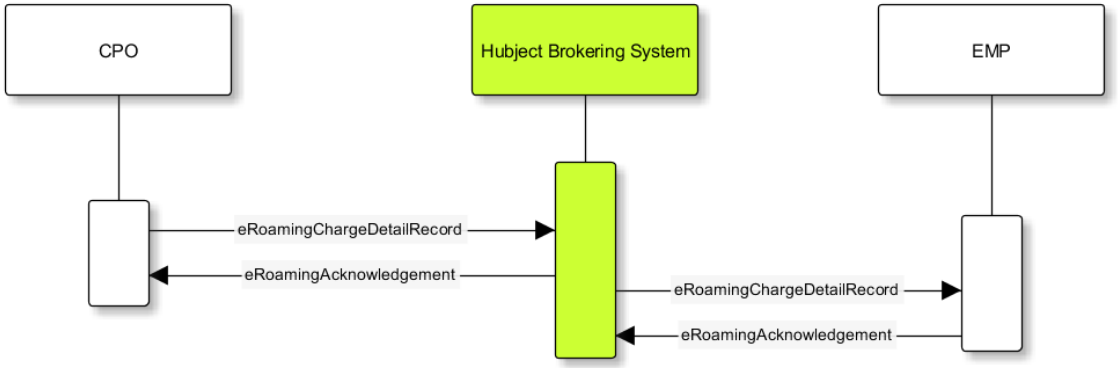


eRoamingAuthorizeRemoteStop basically works in the same way as eRoamingAuthorizeRemoteStart.

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.

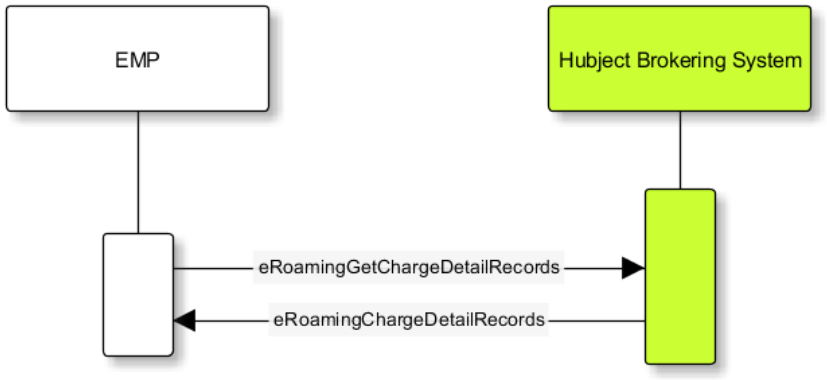
Hubject will identify the receiving EMP and will forward the CDR to the corresponding EMP. The EMP MUST return an eRoamingAcknowledgement message that MUST contain the result indicating whether the session data was received successfully and that MAY contain a status code for further information.

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. If e.g. Hubject was not able to forward a CDR from a CPO to an EMP due to technical problems of the EMP's backend (see 3.1.5), the EMP nevertheless can get access to all CDRs that relate to him. The EMP MUST specify a date range in the request. Hubject will return a list of all CDRs that have been received by Hubject within the specified range and that relate to the requesting EMP (the corresponding charging process was authorized by the EMP or authorized by Hubject based upon the EMP's authentication data).

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

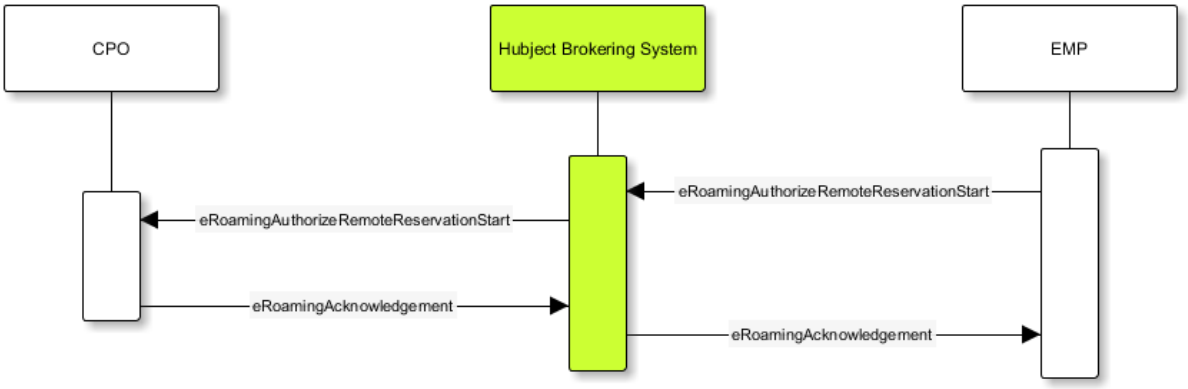
3.2 eRoamingReservation_V1.0

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 to the CPO. If the charging station offers reservations 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 an EMP.

- Request message: eRoamingAuthorizeRemoteReservationStart
- Response message: eRoamingAcknowledgement



Functional description:

Scenario:
A customer of an EMP wants to reserve a charging spot 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 spot 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 spot's compatibility. In case that the CPO has uploaded at least one charging spot 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".

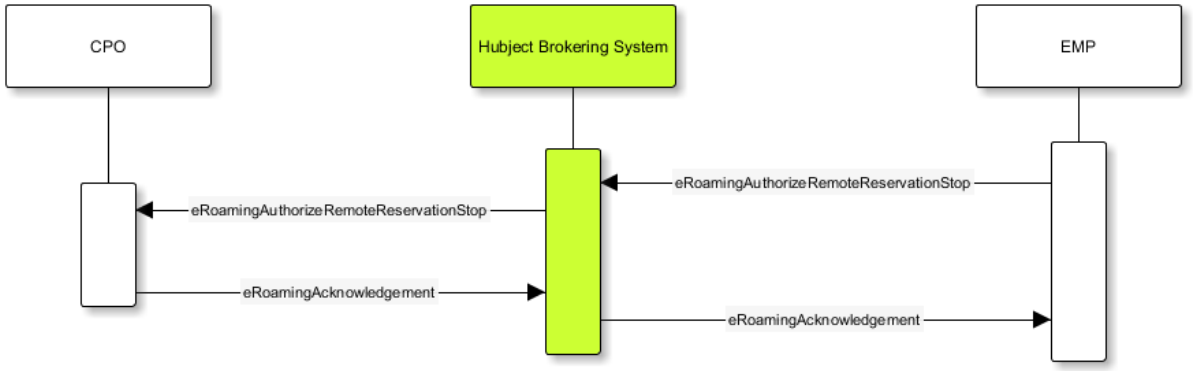
Services and Operations

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_V2.0

The service is only offered by Hubject.

In addition to the online authorization service that requests authentication data on demand and that leaves the customer authentication data completely with 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. This allows for two use cases.

First, in case EMPs upload their data to Hubject, Hubject can authorize requests by partners without having to carry out a further request to EMPs.

The eRoamingPushAuthenticationData gives the EMP the possibility to upload (push) authentication master data.

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

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. The way how Hubject handles the transferred data MUST be defined in the “ActionType” request field (see 3.3.1), which offers four options. The authentication data that will 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, every 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 salted SHA-1 hash using the EVCOID as salt value, adding the salt value to the end of the PIN without delimiter.

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.

Services and Operations

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

3.4 eRoamingEVSEData_V2.1

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

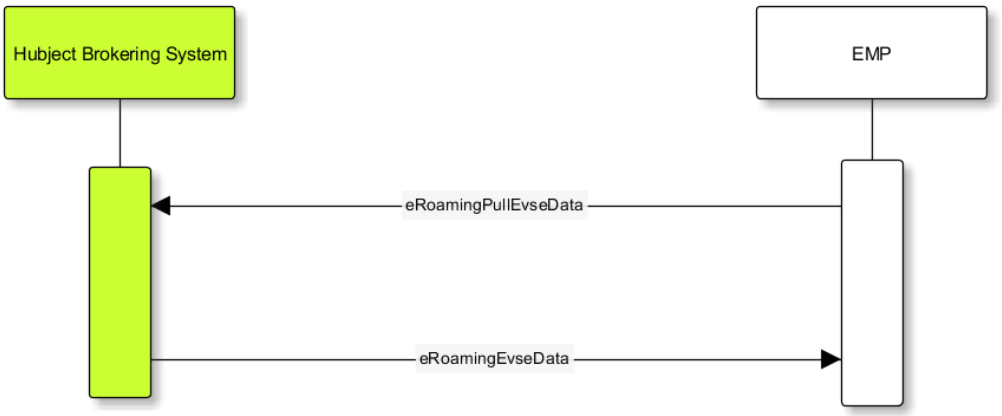
See appendix 6.6 for a detailed business process diagram regarding the EVSE data service.

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. Furthermore, 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 WSDL only allows the provision of either the attribute “SearchCenter” or “LastCall”.

3.5 eRoamingEVSEStatus_V2.0

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

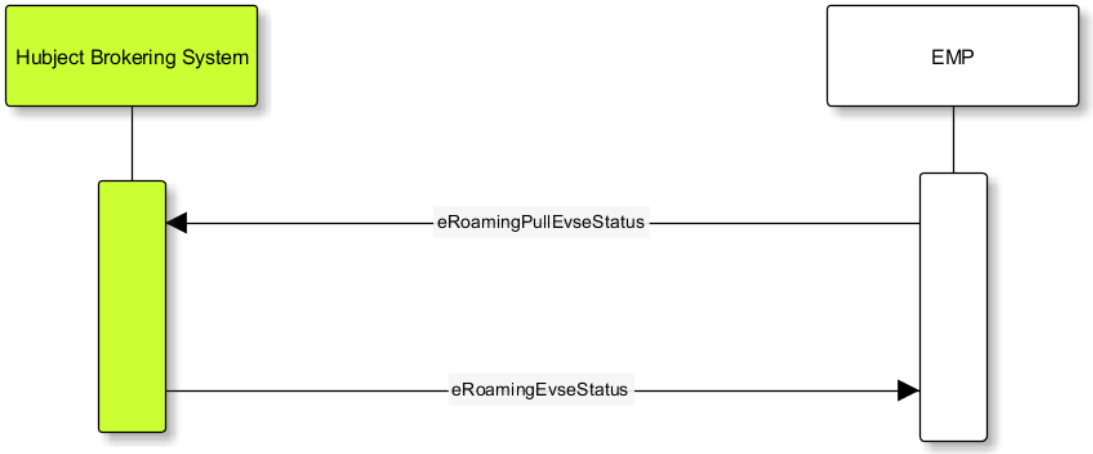
The eRoamingEVSEStatus service offers two operations: eRoamingPushEVSEStatus in order to give CPOs the possibility to upload (push) EVSEStatus data and eRoamingPullEVSEStatus in order to give EMPs the possibility to download (pull) EVSE status data from partner operators via Hubject.

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.

Services and Operations

3.5.1 eRoamingPullEVSEStatus

- Request message: eRoamingPullEVSEStatus
- Response message: eRoamingEVSEStatus

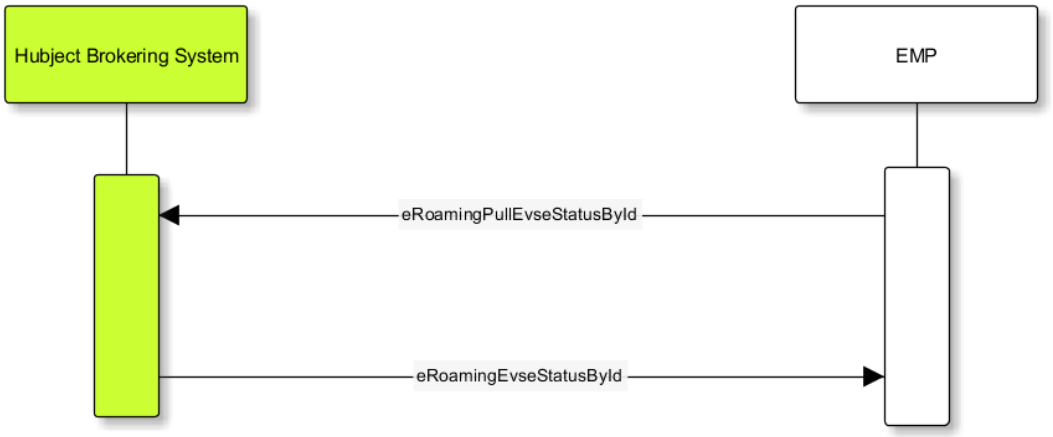


When an EMP sends an eRoamingPullEVSEStatus request, Hubject checks whether there is a valid contract between Hubject and the EMP for the service type (EMP must be the subscriber). If so, the operation allows downloading EVSE status data from Hubject. When an EMP sends an eRoamingPullEVSEStatus request, 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. The operation also serves as interface for EMPs in order to download EVSE status records. In contrast to the eRoamingPullEVSEStatus operation the eRoamingPullEVSEStatusById operation requires a list of EVSEIDs sent with the request message.

Services and Operations

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 EVSE.

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 send messages by EMP

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 who implement the offline EMP.

EMP → HBS		**Related Service Version: V_2.0**	
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 sending of EVSE data of operators and that is stored on the Hubject system.

EMP → HBS		**Related Service Version: V_2.1**	
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”.	O
GeoCoordinatesResponseFormat	GeoCoordinates ResponseFormatType (see 5.2.8)	Defines the format of geo coordinates that shall be provided with the response.	M

Messages

4.1.3 eRoamingPullEVSEStatus

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

EMP → HBS	**Related Service Version: V_2.0**
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 5.2.10)	Status of the EVSE	O

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

Messages

4.1.4 eRoamingAuthorizeRemoteStart

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

EMP → HBS		**Related Service Version: V_2.0**		
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	
PartnerSessionID	String	Partner systems can use this field in order to relate the process to their own session handling.	O	50
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	String	Service specifics like e.g. “fast charging”.	O	100

Messages

4.1.5 eRoamingAuthorizeRemoteStop

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

EMP → HBS		**Related Service Version: V_2.0**		
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	
PartnerSessionID	String	Partner systems can use this field in order to relate the process to their own session handling.	O	50
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_2.0**	
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 send in return to the eRoamingAuthorizeStart and is only mandatory for online EMPs.

EMP → HBS		**Related Service Version: V_2.0**		
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	
PartnerSessionID	String	If a partner session ID has been provided with the start request, it is added here to the response.	O	50
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	
AuthorizationStop Identifications	List (IdentificationType) (see 5.1.2)	A list of Identification data that is authorized to stop the charging process.	O	

Messages

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 send in return to the eRoamingAuthorizeStop and is only mandatory for online EMPs.

EMP → HBS		**Related Service Version: V_2.0**		
Request: eRoamingAuthorizeStop (see 4.2.3)		**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.	M	
PartnerSessionID	String	Partner systems can use this field to relate the process to their own session handling.	O	50
OperatorID	OperatorIDType (see 5.3.4)	The OperatorID is defined by Hubject and is used to identify the CPO.	M	
Identification	IdentificationType (see 5.1.2)	Authentication data used to authorize the user or car.	M	

Messages

4.2 Mandatory received messages by EMP

4.2.1 eRoamingChargeDetailRecord

eRoamingChargeDetailRecord is a message informing about charging process details (e.g. meter values).

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

HBS → EMP		**Related Service Version: V_2.0**		
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	
PartnerSessionID	String	Partner systems can use this field to relate the process to their own session handling.	O	50
PartnerProductID	String	Service specifics like e.g. “fast charging”.	O	100
EVSEID	EVSEIDType (see 5.3.2)	The ID that identifies the charging spot.	M	
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	M	

Messages

HBS → EMP		**Related Service Version: V_2.0**		
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**
		RFID or cable connected.		
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	
HubProviderID	ProviderIDType (see 5.3.3)	Hub provider	O	

Messages

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 in return and is only mandatory for online EMPs.

HBS → EMP		**Related Service Version: V_2.0**		
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	
PartnerSessionID	String	Partner systems can use this field to relate the process to their own session handling.	O	50
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	String	Service specifics like e.g. “fast charging”.	O	100

Messages

4.2.3 eRoamingAuthorizeStop

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

HBS → EMP		**Related Service Version: V_2.0**		
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	
PartnerSessionID	String	Partner systems can use this field to relate the process to their own session handling.	O	50
OperatorID	OperatorIDType (see 5.3.4)	The OperatorID is defined by Hubject and is used to identify the CPO.	M	
Identification	IdentificationType (see 5.1.2)	Authentication data used to authorize the user or car.	M	

Messages

4.2.4 eRoamingEVSEData

eRoamingEVSEData is sent in response to eRoamingPullEVSEData requests.

Please note:
This message describes the response which has to be received in return to the eRoamingPullEVSEData.

HBS → EMP		**Related Service Version: V_2.1**	
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 in return to the eRoamingPullEVSEStatus.

HBS → EMP		**Related Service Version: V_2.0**	
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

Messages

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	
PartnerSessionID	String	In case that a partner SessionID has been provided with the start request, it is added here to the response.	O	50

Messages

4.3 Optional send messages by EMP

4.3.1 eRoamingPullEVSEStatusByID

eRoamingPullEVSEStatusByID is a message that is sent in order to request the EVSE status according to specific EVSE ID.

EMP → HBS		**Related Service Version: V_2.0**	
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

Messages

4.3.2 eRoamingAuthorizeRemoteReservationStart

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

EMP → HBS		**Related Service Version: V_1.0**		
Request: **eRoamingAuthorizeRemoteReservationStart**		**Response:** **eRoamingAcknowledgement (see 4.2.6)**		
Direction: **EMP MAY send message from 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	
PartnerSessionID	String	Partner systems can use this field in order to relate the process to their own session handling.	O	50
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	String	Service specifics like e.g. “15min”.	O	100

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_1.0**		
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	
PartnerSessionID	String	Partner systems can use this field in order to relate the process to their own session handling.	O	50
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

4.4 Optional received messages by EMP

4.4.1 eRoamingEVSEStatusByID

eRoamingEVSEStatusById is sent in response to eRoamingPullEVSEStatusById requests.

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

HBS → EMP		**Related Service Version: V_2.0**		
Request: **eRoamingPullEVSEStatusByID (see 4.3.1)**		**Response:** **eRoamingEVSEStatusByID**		
Direction: **EMP MAY send 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	

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 5.2.11)	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.	M One of the four options MUST be provided.
QRCodeIdentification	QRCodeIdentificationType (see 5.1.11)		
PlugAndChargeIdentification	PlugAndChargeIdentificationType (see 5.1.12)		
RemoteIdentification	RemoteIdentificationType (see 5.1.13)		

Data Types

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)		

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

Data Types

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 (see 5.1.2)	Authentication data.	M

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

Data Types

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

Data Types

5.1.13 RemoteIdentificationType

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 5.2.12)	M	
City	String	M	1-50
Street	String	M	5-100
PostalCode	String	O	10
HouseNum	String	O	10
Floor	String	O	5
Region	String	O	50
Timezone	String	O	10

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

Data Types

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	
ChargingStationId	String	The ID that identifies the charging station.	O	50
ChargingStationName	String	Name of the charging station.	O	50
EnChargingStationName	String	Name of the charging station in English.	O	50

Data Types

Name	Data Type	Description	M/O	Field length
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 5.2.4)	List of plugs that are supported.	M	
ChargingFacilities	List (ChargingFacilityType) (see 5.2.3)	List of facilities that are supported.	O	
ChargingModes	List (ChargingModeType) (see 5.2.5)	List of charging modes that are supported.	O	
AuthenticationModes	List (AuthenticationModeType) (see 5.2.6)	List of authentication modes that are supported.	M	
MaxCapacity	Integer	Maximum capacity in kWh.	O	
PaymentOptions	List (PaymentOptionType) (see 5.2.7)	List of payment options that are supported.	O	
ValueAddedServices	List ValueAddedServiceType (see 5.2.14)	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	
AdditionalInfo	String	Optional information.	O	200
EnAdditionalInfo	LocalizedInfoType (see 5.3.12)	Optional information.	O	2000

Data Types

Name	Data Type	Description	M/O	Field length
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	
OpeningTime	String	Opening time in case that the charging station cannot be accessed around the clock.	O	200
HubOperatorID	OperatorIDType (see 5.3.4)	Hub operator.	O	3
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	Does the operator send a EVSEStatus information for this EVSERecord? Values: true / false / auto Value auto is set to true by Hubject if the operator offers Hubject EVSEData.	M	

Data Types

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 5.1.17)	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 5.2.10)	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.	O	100

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.

5.2.3 ChargingFacilityType

Option	Description
100 - 120V, 1-Phase ≤ 10A	Defined charging facility.
100 - 120V, 1-Phase ≤ 16A	Defined charging facility.
100 - 120V, 1-Phase ≤ 32A	Defined charging facility.
200 - 240V, 1-Phase ≤ 10A	Defined charging facility.
200 - 240V, 1-Phase ≤ 16A	Defined charging facility.
200 - 240V, 1-Phase ≤ 32A	Defined charging facility.
200 - 240V, 1-Phase > 32A	Defined charging facility.
380 - 480V, 3-Phase ≤ 16A	Defined charging facility.

Data Types

Option	Description
380 - 480V, 3-Phase ≤ 32A	Defined charging facility.
380 - 480V, 3-Phase ≤ 63A	Defined charging facility.
Battery exchange	Defined charging facility.
DC Charging ≤ 20kW	Defined charging facility.
DC Charging ≤ 50kW	Defined charging facility.
DC Charging > 50kW	Defined charging facility.
Unspecified	Defined charging facility.

5.2.4 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.
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.

Data Types

Option	Description
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.

5.2.5 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.

5.2.6 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*

Data Types

5.2.7 PaymentOptionType

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

5.2.8 GeoCoordinatesResponseFormatType

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

5.2.9 HashFunctionType

Option	Description
MD5	Hash value is based on MD5.
SHA-1	Hash value is based on SHA-1.

5.2.10 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.

5.2.11 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 underlying EVCOID.	Authentication codes
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

Data Types

Option	Description	Area of usage
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

5.2.12 CountryCodeType

The CountryCodeType allows for Alpha-3 as well as Alpha-2 country codes.

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.

For Alpha-2 (two-letter) country codes as defined in ISO 3166-1 see http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2 for the full code list.

Examples:

Option	Description
AUT	Austria
DEU	Germany
FRA	France
USA	United States
AT	Austria
DE	Germany
FR	France
US	United States

5.2.13 ProductIDType

The ProductIDType defines standard values (see below). Nevertheless the type allows for providing custom product specifications (String with 100 characters maximum length).

Option	Description
Standardpreis	Standard price
AC1	
AC3	

Data Types

Option	Description
DC	
ProductIDType	There is no option “ProductIDType”, this only indicates, that custom product specifications (String with 100 characters maximum length) are allowed, also.

5.2.14 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.

Data Types

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|X])|
([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 on OICP 2.0 which are using the ISO Standard of EvcoID’s we recommend to use the following marked characters whithin the EvcoId DE-ABC-C123456QQ-9 For further information please check the OICP mapping documentation on page 7

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}
```

Data Types

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.

Data Types

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 LocalizedInfoType

([A-Z]{3}:.*|||)+

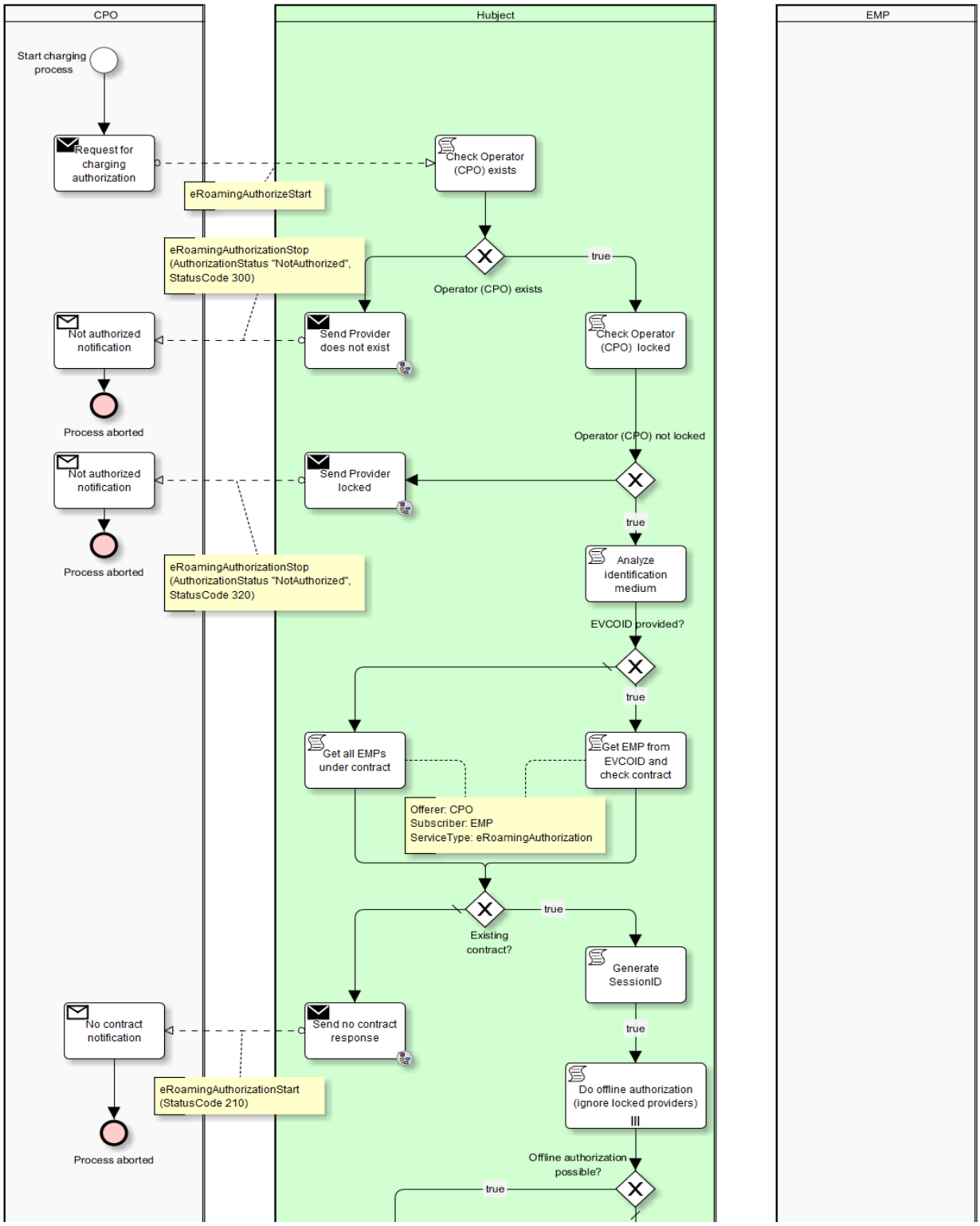
The expression validates the string as a structure, which contains one or more language specific information. Every language specific section must start with a three-letter Alpha-3 country code (ISO 3166-1) followed by “:”. The section must end with the separator string “|||”.

Example: “DEU:Inhalt|||GBR:Content|||FRA:Objet|||”

The LocalizedInfoType is used in the EnAdditionalInfo field, which is part of the EVSEDataRecordType (see 5.1.17).

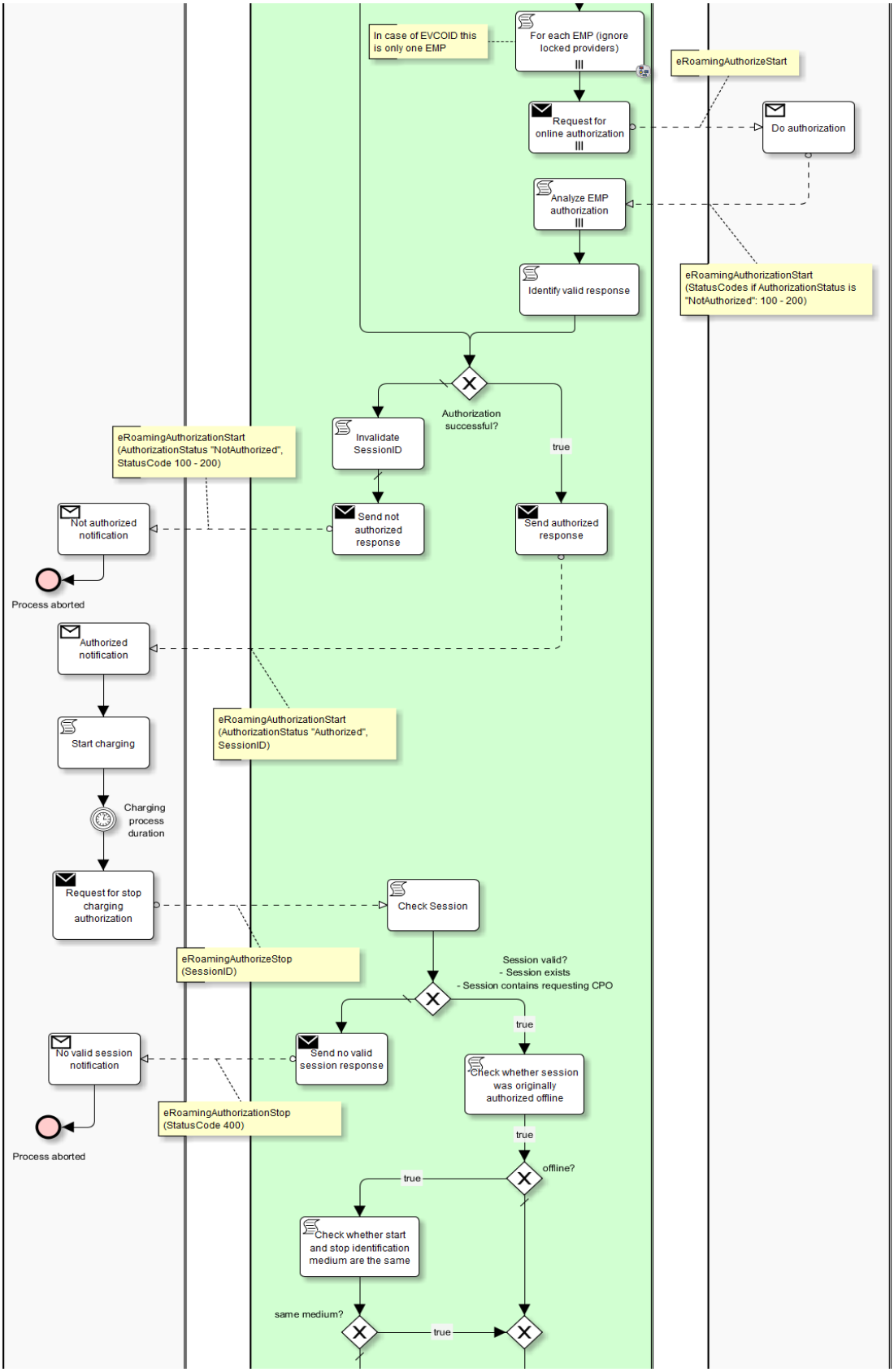
6 Appendix

6.1 Business Process Diagram eRoamingAuthorization



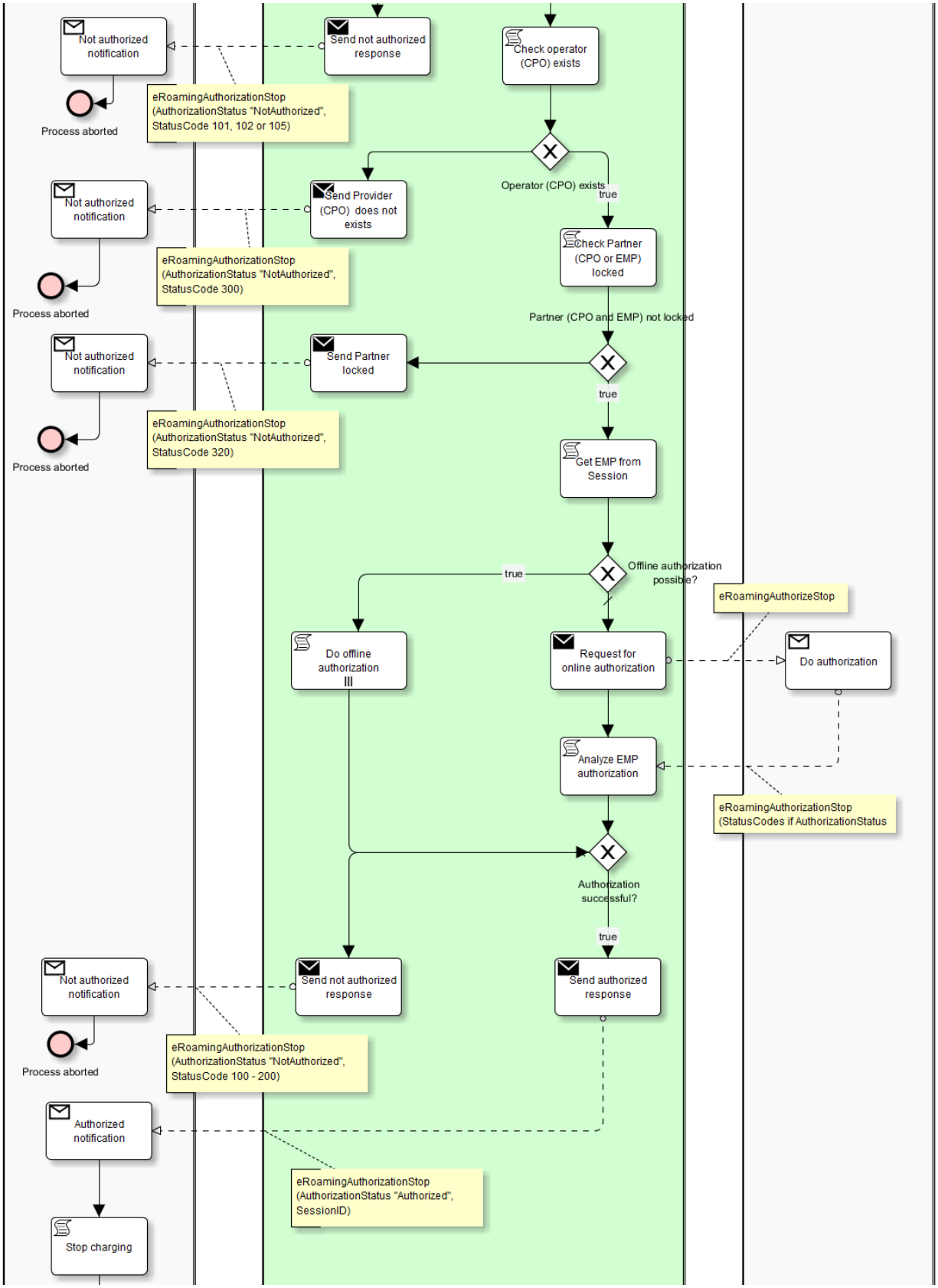
Appendix

Business Process Diagram eRoamingAuthorization



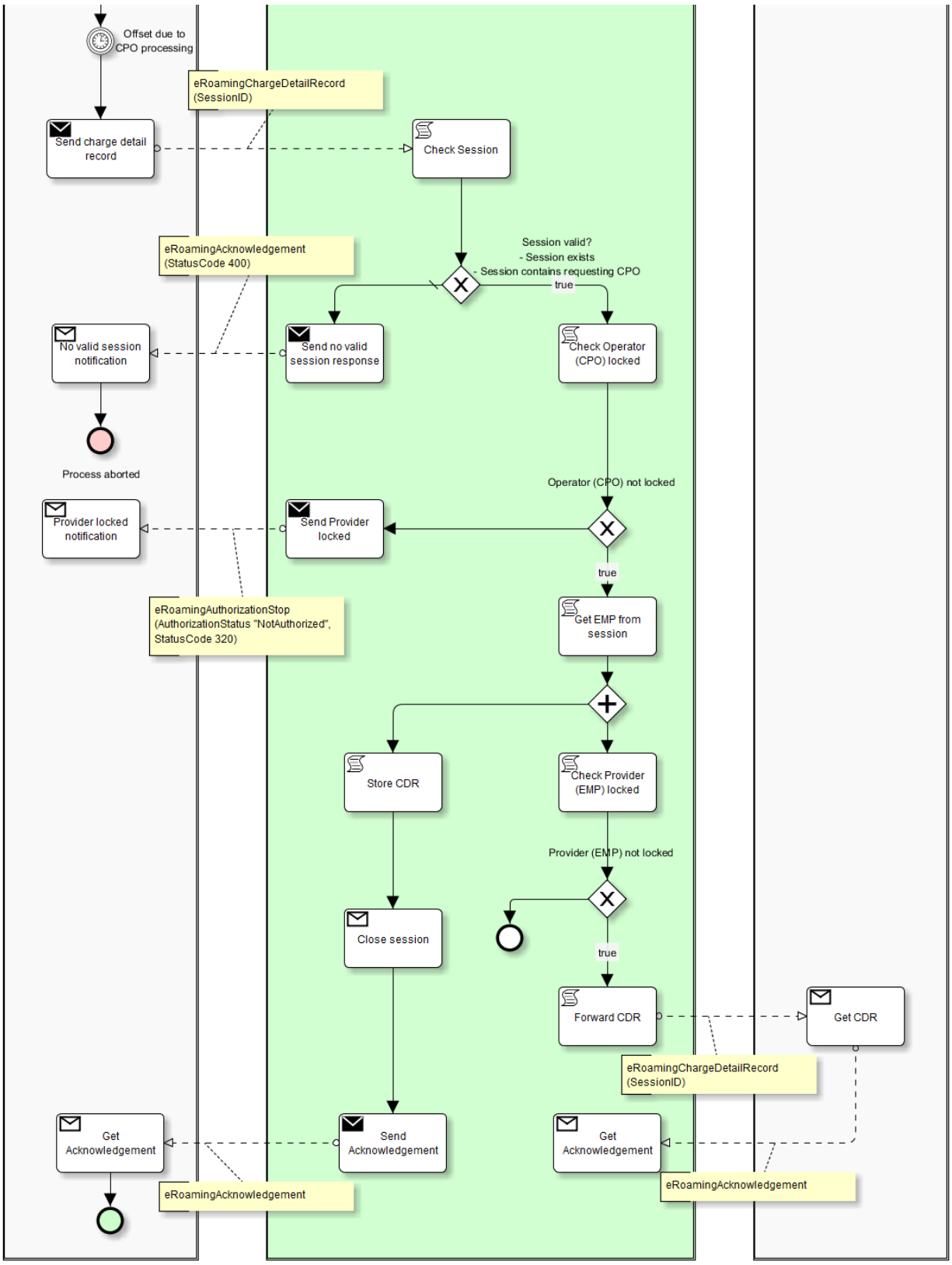
Appendix

Business Process Diagram eRoamingAuthorization



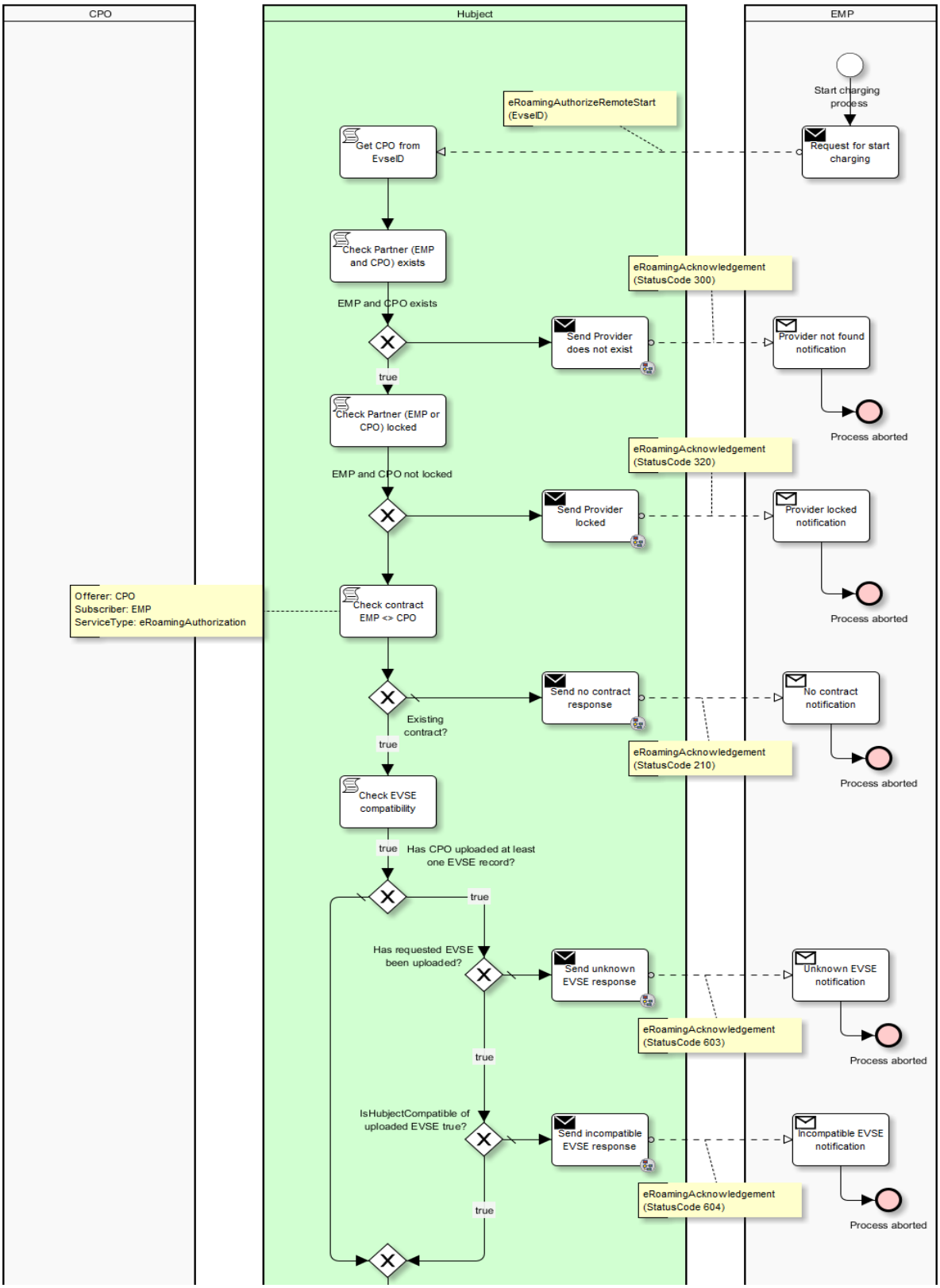
Appendix

Business Process Diagram eRoamingAuthorization



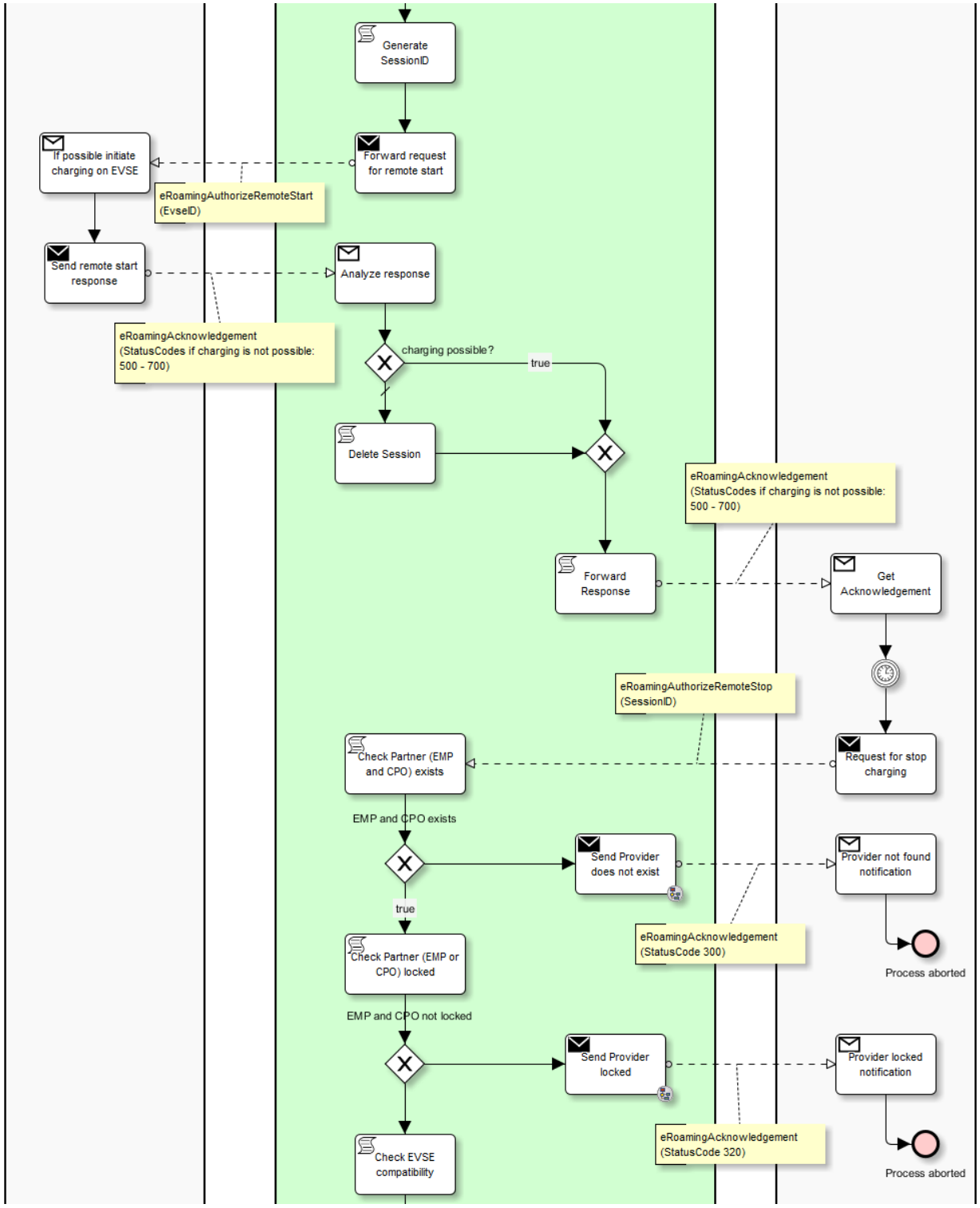
Appendix

6.2 Business Process Diagram eRoamingAuthorization Remote



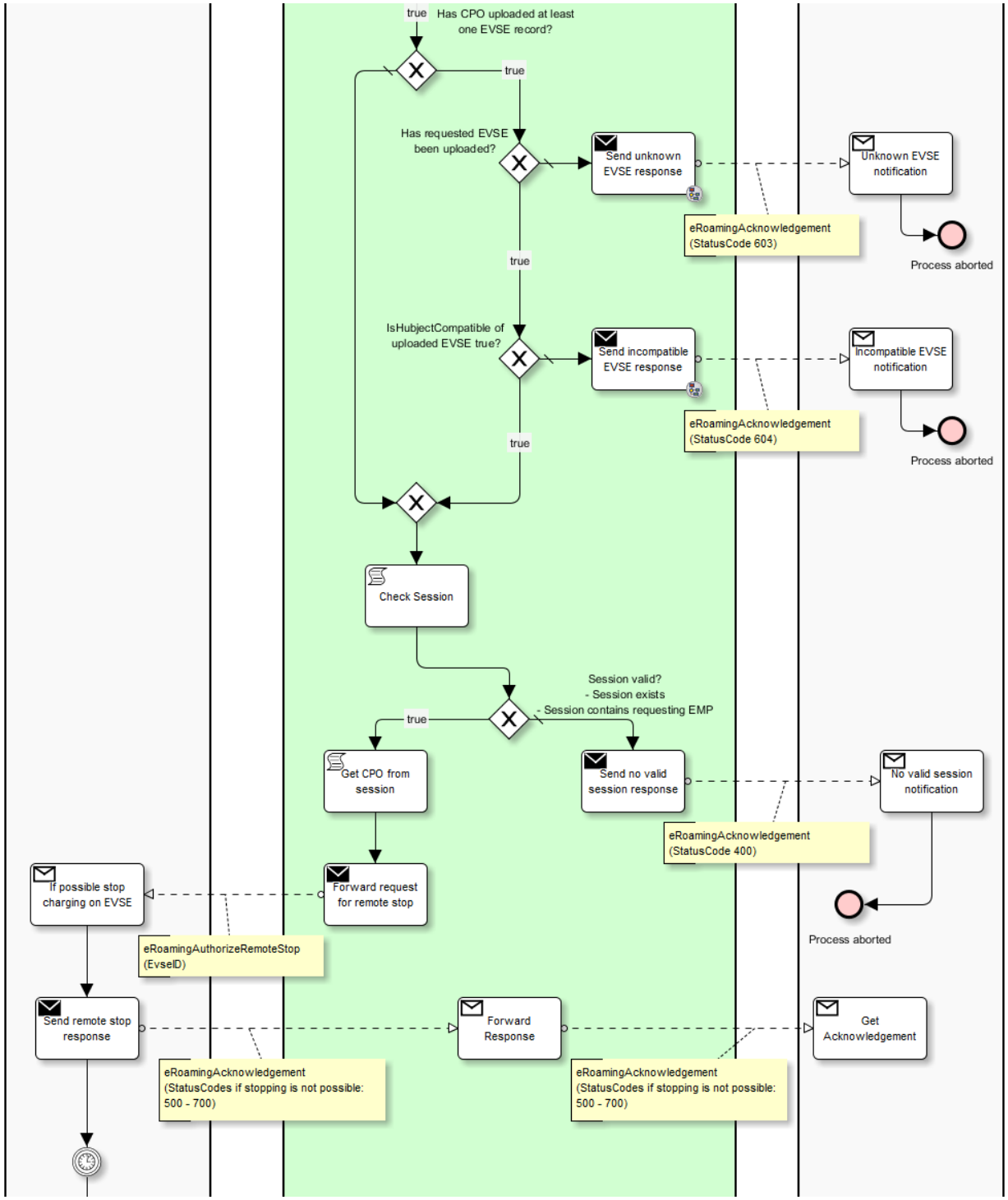
Appendix

Business Process Diagram eRoamingAuthorization Remote



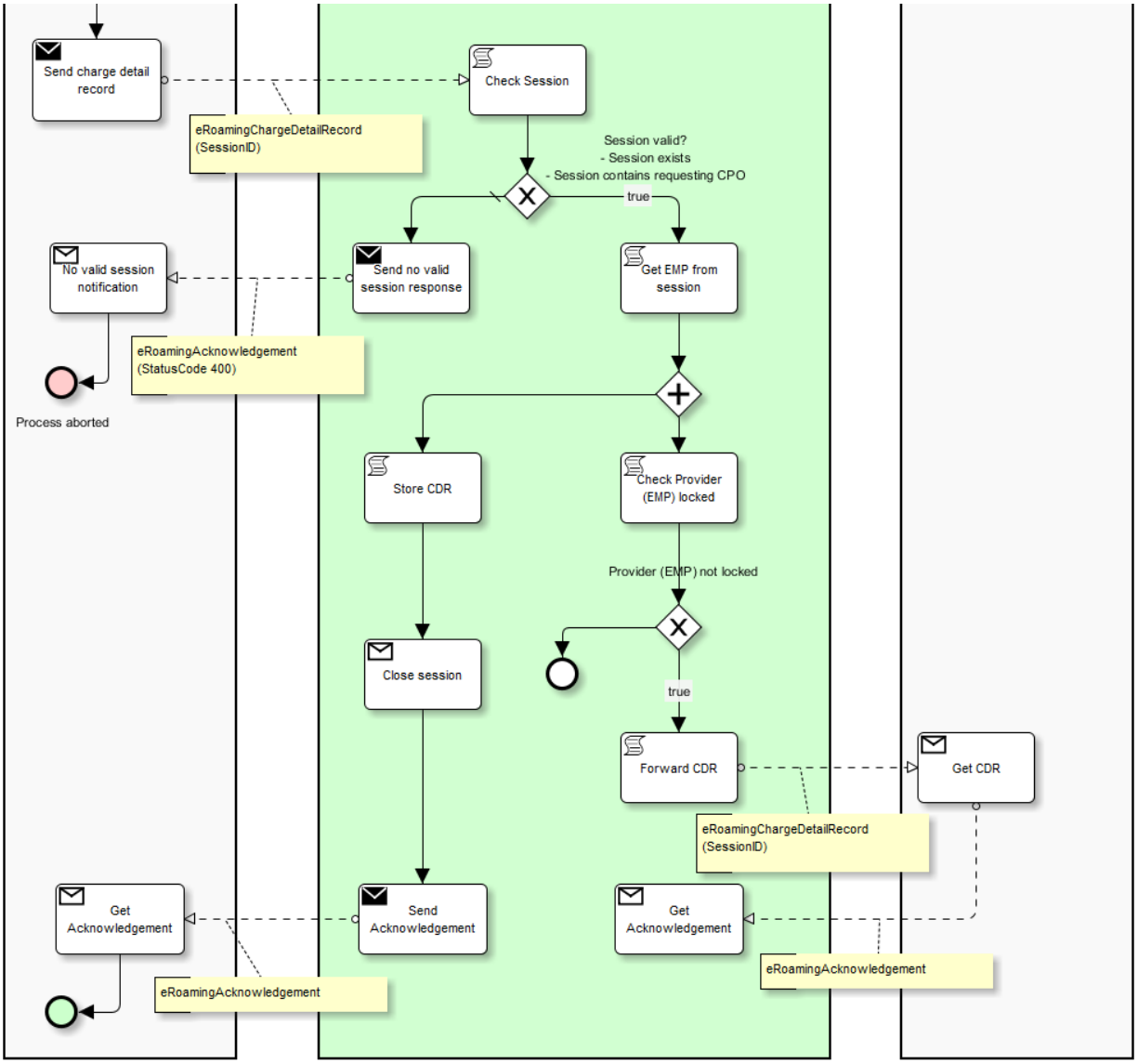
Appendix

Business Process Diagram eRoamingAuthorization Remote

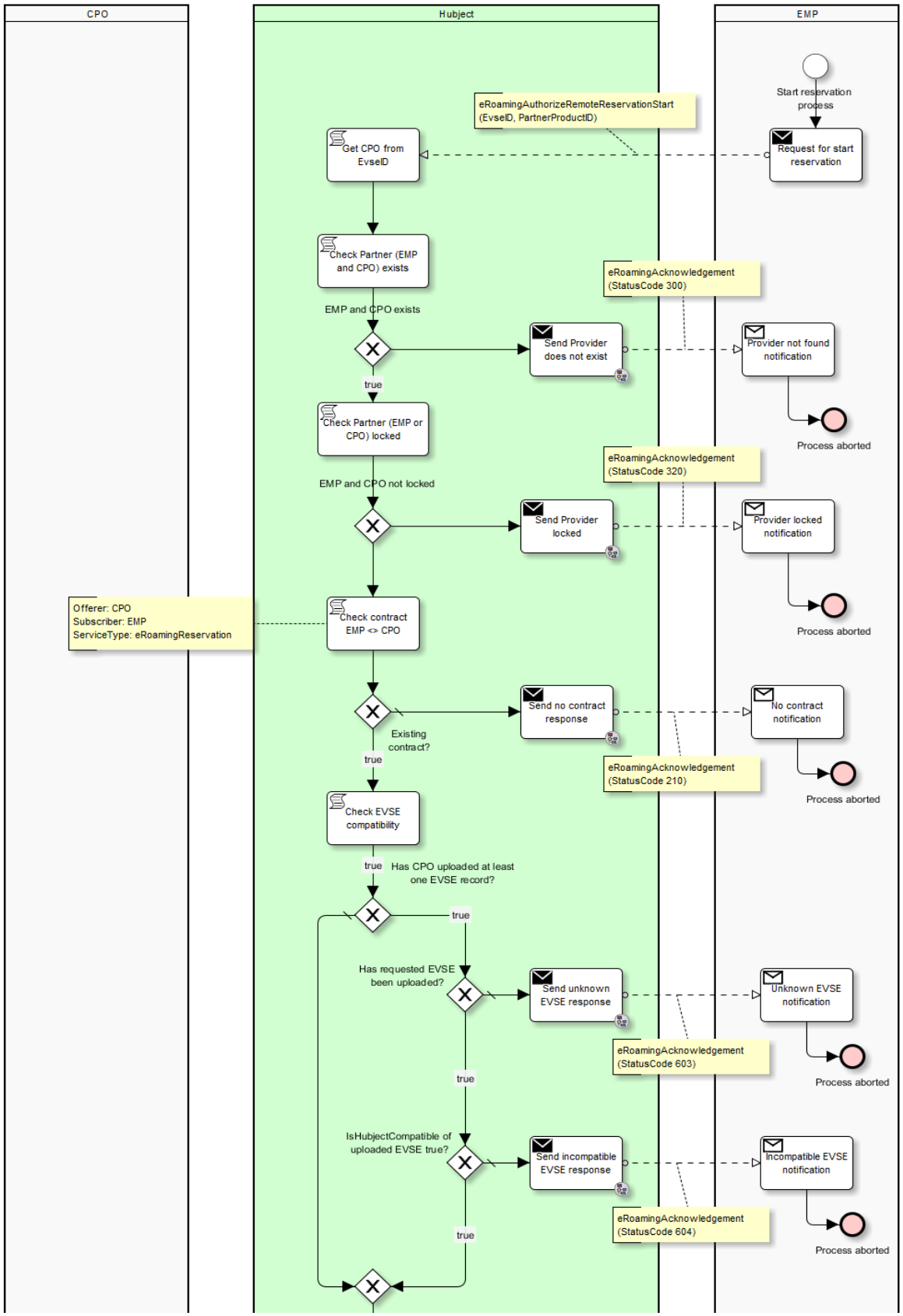


Appendix

Business Process Diagram eRoamingAuthorization Remote

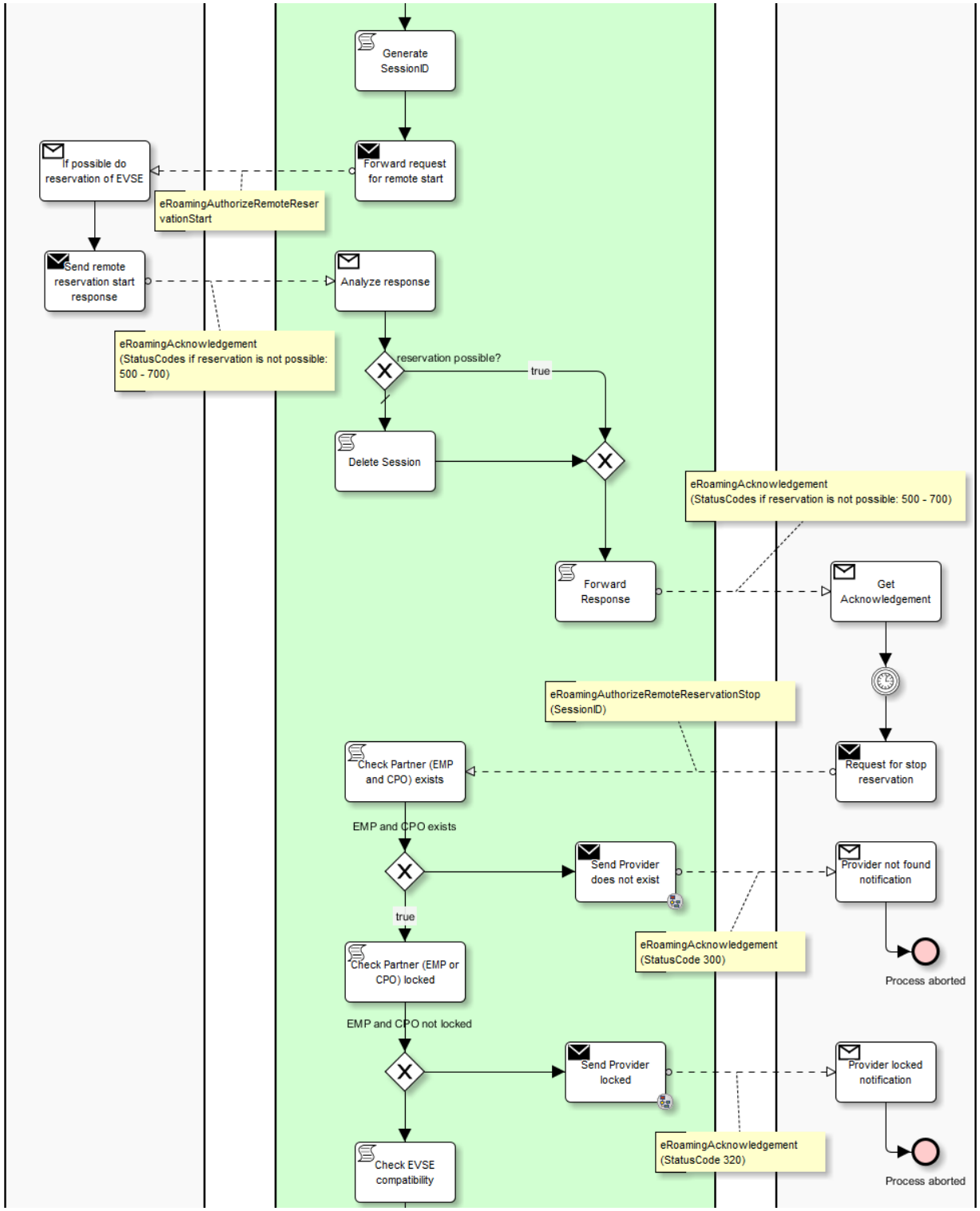


6.3 Business Process Diagram eRoamingReservation



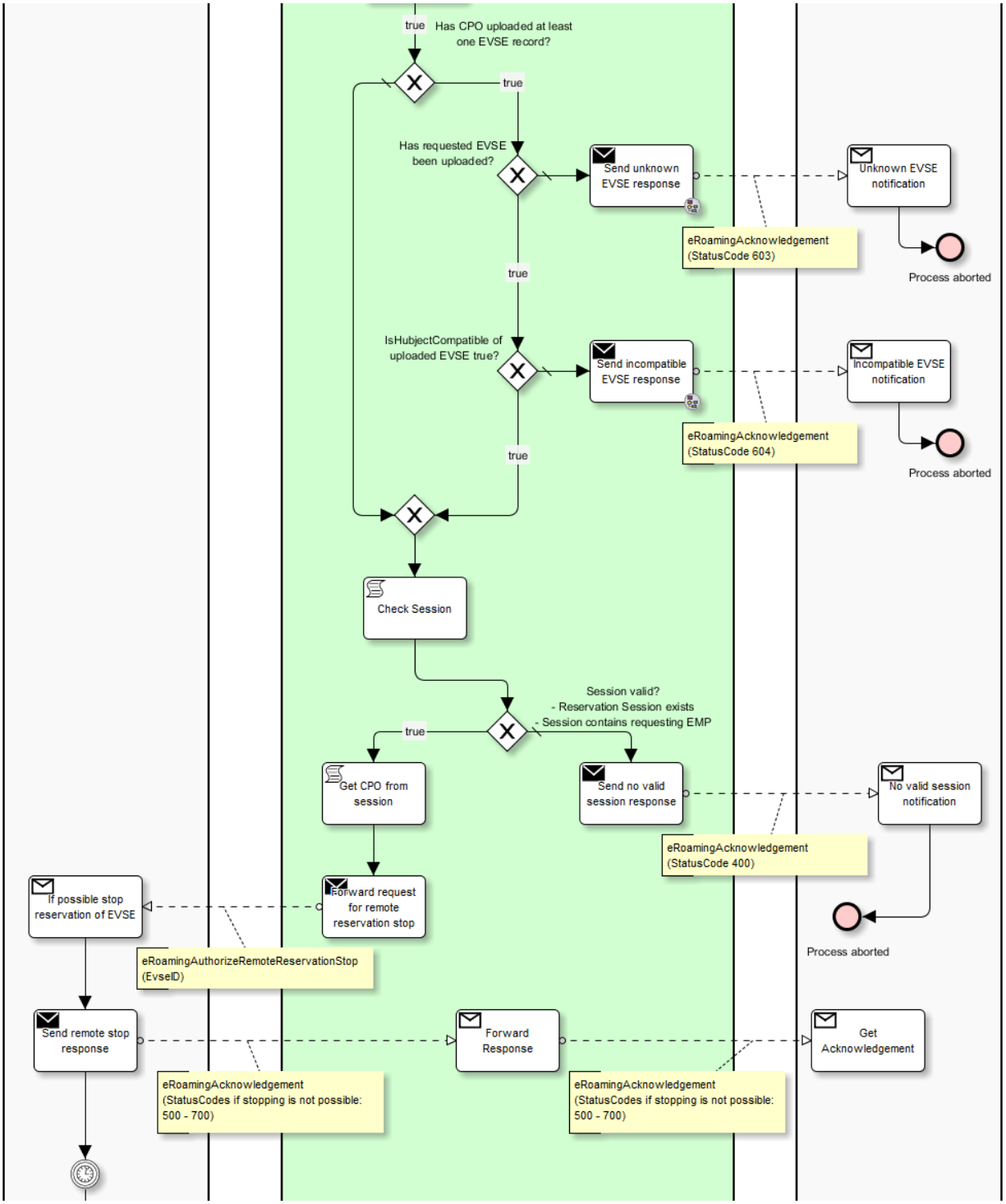
Appendix

Business Process Diagram eRoamingReservation



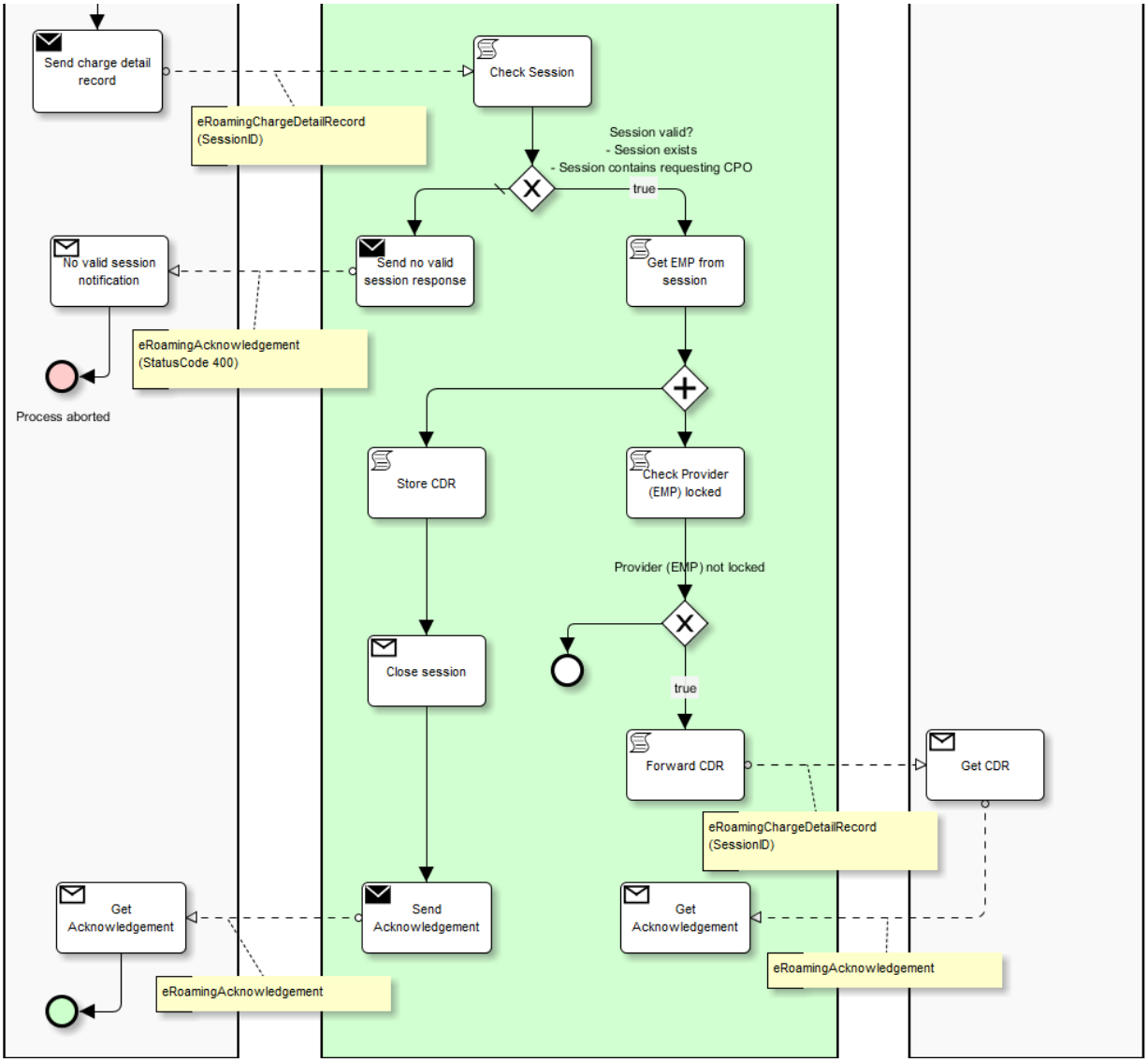
Appendix

Business Process Diagram eRoamingReservation



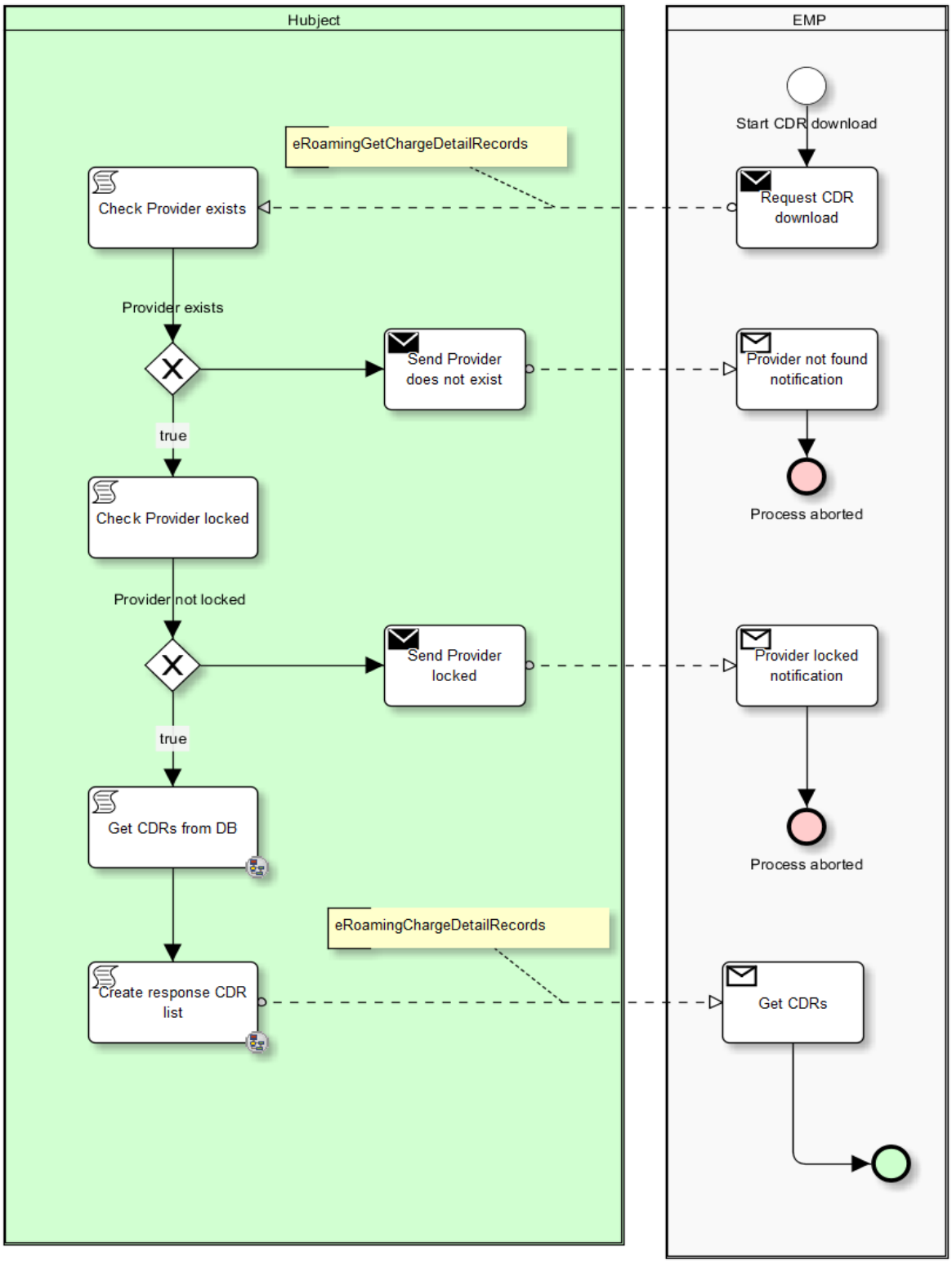
Appendix

Business Process Diagram eRoamingReservation



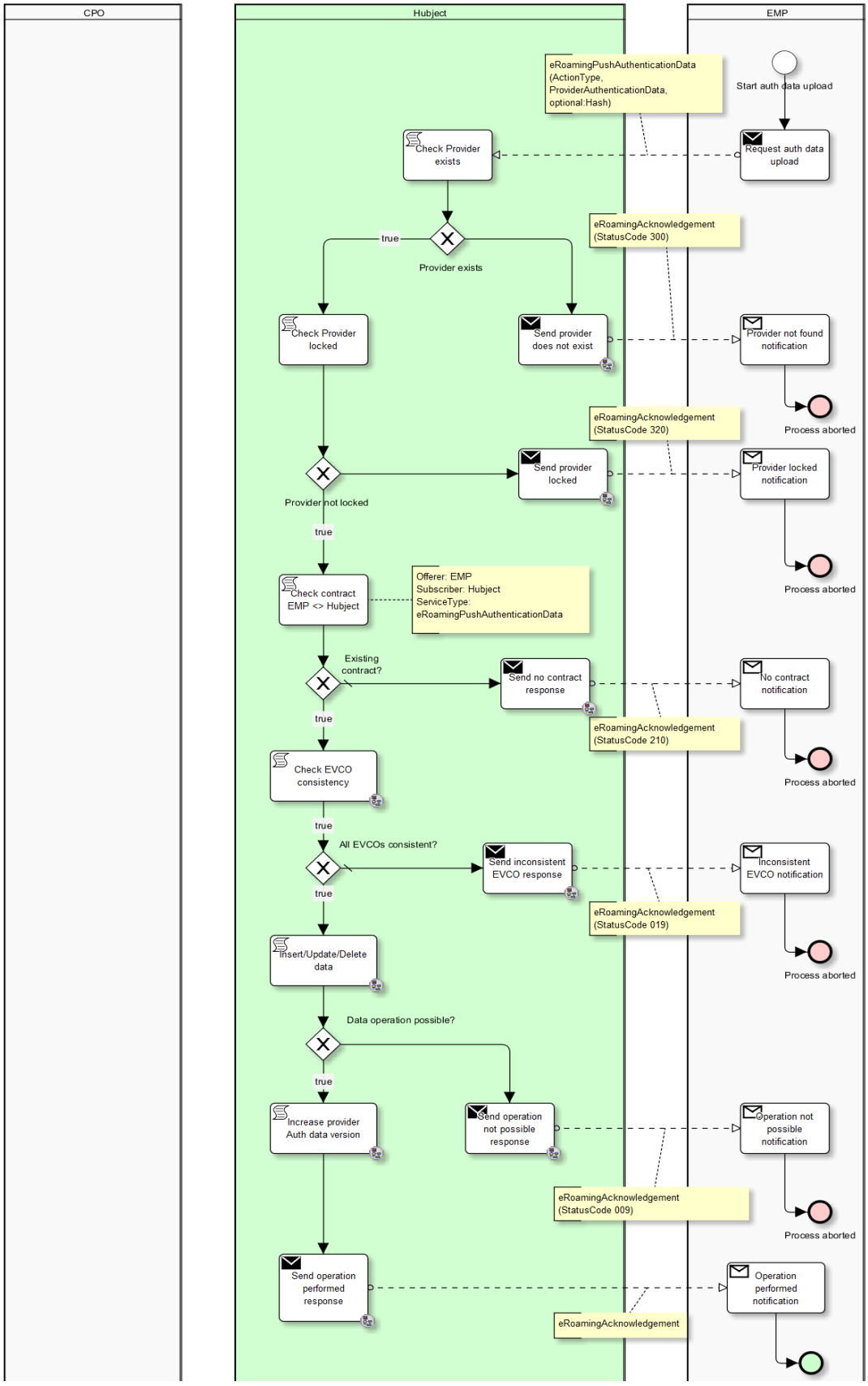
Appendix

6.4 Business Process Diagram eRoamingAuthorization GetCDRs



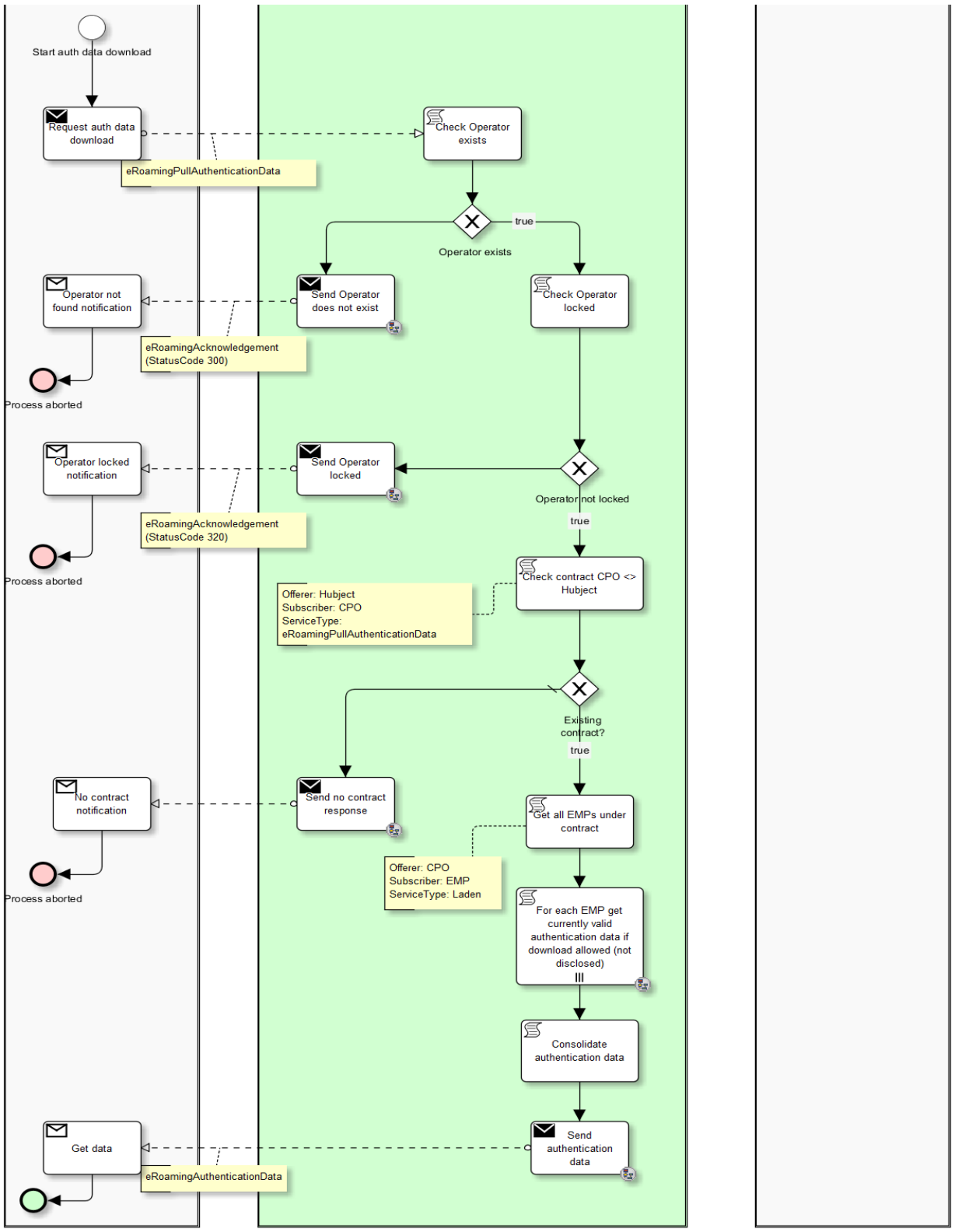
Appendix

6.5 Business Process Diagram eRoamingAuthenticationData

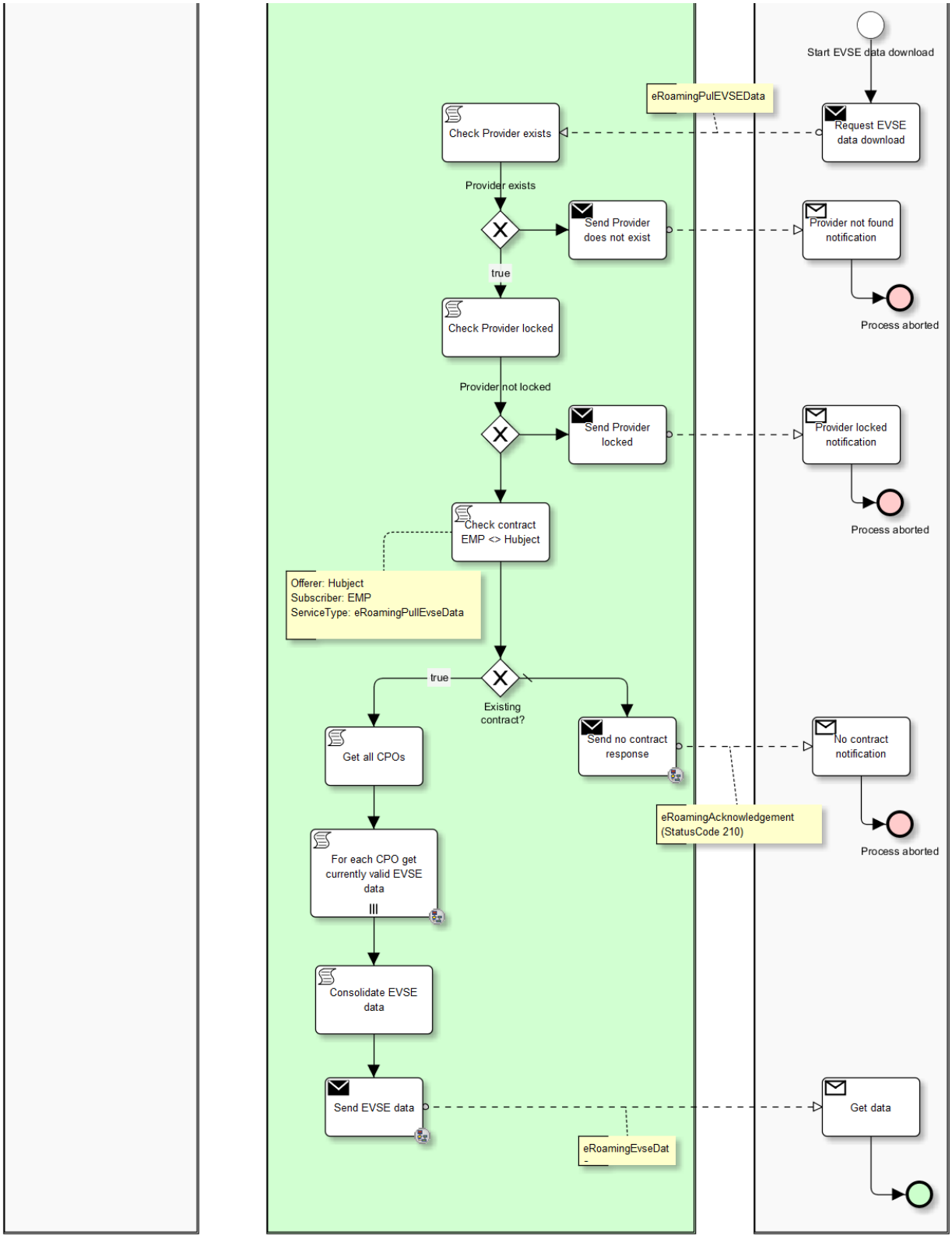


Appendix

Business Process Diagram eRoamingAuthenticationData

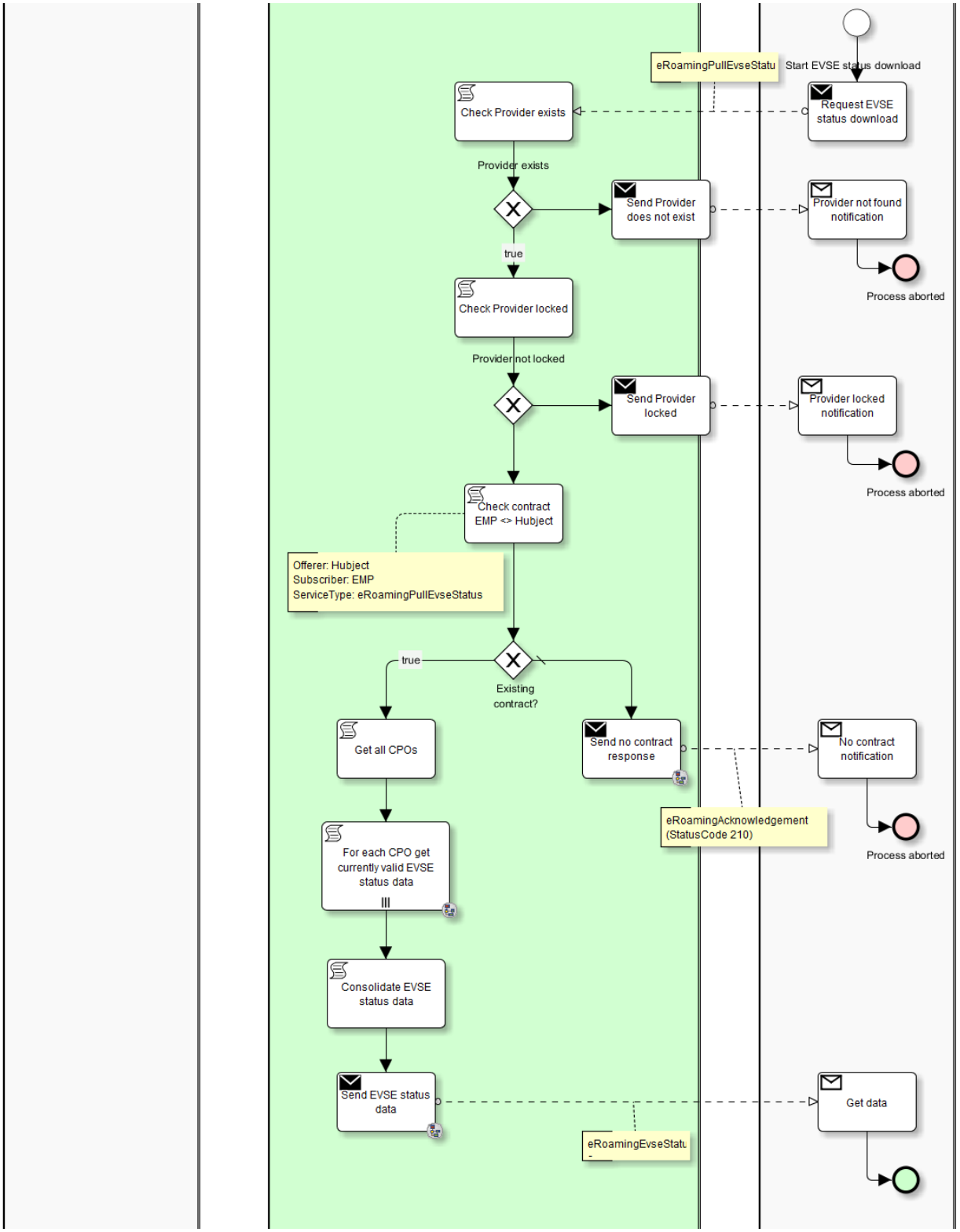


6.6 Business Process Diagram eRoamingEVSEData



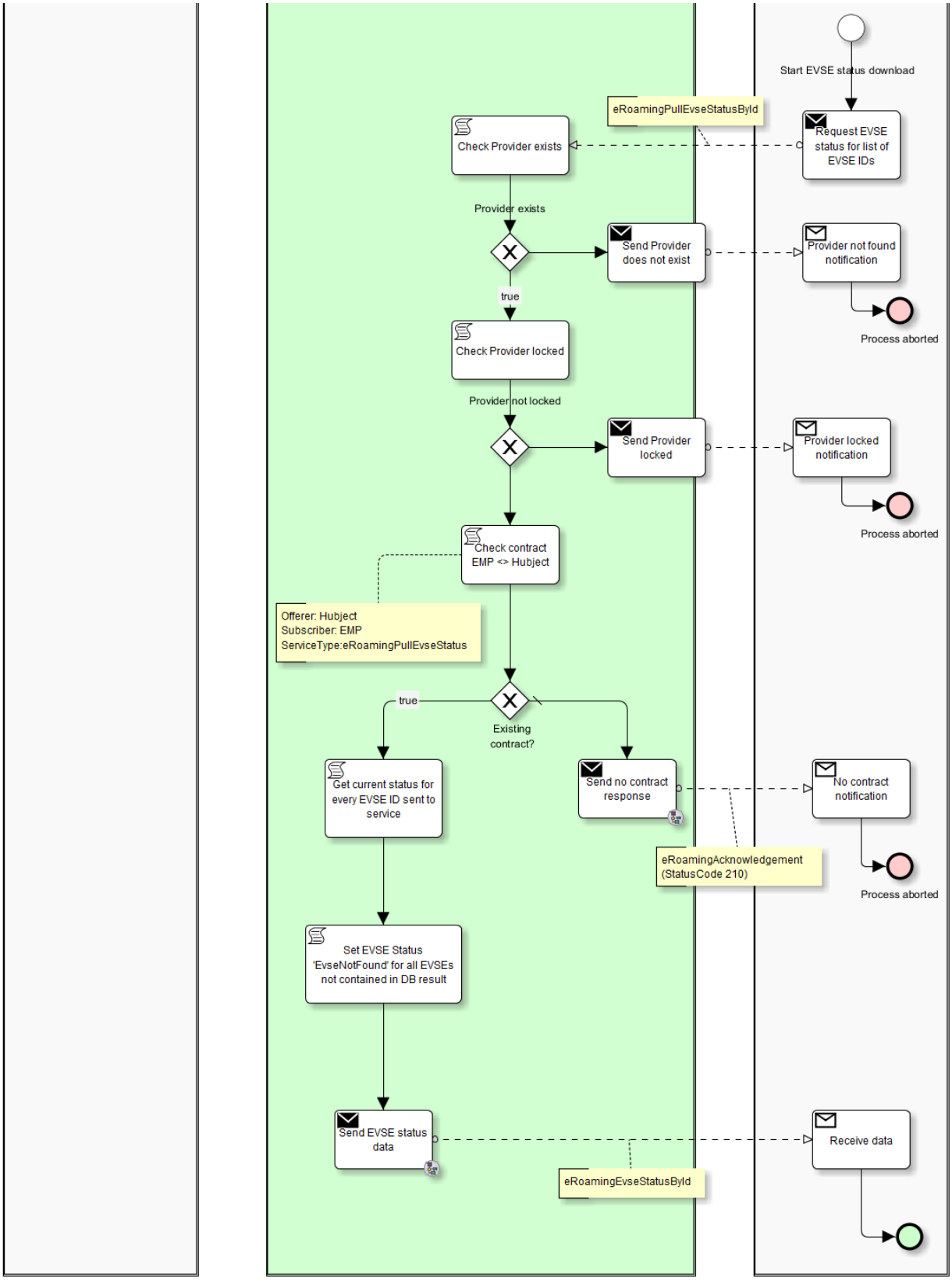
Appendix

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.