OSCP 2.0 - Specification

FINAL, 2020-10-12

Table of Contents

Disclaimer

1. Introduction

1.1. Goal

The goal of this document is to describe a protocol for using flexible energy resources based on available capacity, primarily aimed at smart charging electrical vehicles (EVs) based on available capacity. This document describes use cases in which the messages are applied in more generic terms than OSCP 1.0, which was specifically aimed at smart charging of electric vehicles by a Distribution System Operator (DSO). The reason for using more generic terms is that this specification does not want to limit possibilities of the protocol to smart charging EVs. This is driven by the integration of EVs in larger energy ecosystems, including PV, stationary batteries, heat pumps and other devices.

1.2. Scope

This document describes the use cases and messages defined for using flexible energy resources based on available capacity, the primary focus on capacity based smart charging. The scope of this document is primarily focused on the communication between the system of a *Capacity Provider*, for example a DSO, and a system that can offer flexibility, for example a Charging Station Management System of a Charging Station Operator. The latter will in this document be referred to as a *Flexibility Provider*.
The communication between the Flexibility Provider and the resources it uses to provide flexibility, *Flexibility Resources*, are out of scope of this document. In a previous version of OSCP this was addressed, since this functionality was not publicly available yet. Since this is not the case anymore (see for example the latest version of OCPP), this part of the communication is left out of scope of this version of OSCP.

1.3. Conventions

1.3.1. Normative

All sections and appendices are normative, unless they are explicitly indicated to be informative.

1.3.2. Requirement Keywords

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 RFC2119, subject to the following additional clarification clause:

The phrase "valid reasons in particular circumstances" relating to the usage of the terms "SHOULD", "SHOULD NOT", "RECOMMENDED", and "NOT RECOMMENDED" is to be taken to mean technically valid reasons, such as the absence of necessary software to support a function from a system: for the purposes of this specification it specifically excludes decisions made on commercial, or other non-technical grounds, such as cost of implementation, or likelihood of use.

1.3.3. Datatypes

The specification mentions the following datatypes:

Table 1. Datatypes

Datatype	Description
string	The characters defined in the Unicode character set are allowed to be used.
object	An unordered collection of key:value pairs.
integer	32 bit (31 bit resolution, 1 sign bit) No leading 0s No plus sign Allowed value examples: 1234, -1234 Not Allowed: 01234, +1234
decimal	A decimal to use in OSCP is a floating point number with a maximum of 8 decimal places.
datetime	All time values exchanged between CSMS and Charging Station SHALL be formatted as defined in [RFC3339]. Additionally fractional seconds have been given an extra limit. The number of decimal places SHALL NOT exceed the maximum of 3.
AnyType	Text, data without specified length or format.
boolean	Only allowed values: "false" and "true".
URL	A string of max 255 characters that represents a web address, as defined in URL specification
null	definition

1.4. Terminology

This section contains the terminology that is used throughout this document.

Table 2. Terminology

Terminology	Description
Application layer	OSI-Layer 5-7.
Group capacity	This is the maximum capacity (in Amps) that can go to a group before something goes wrong (burning a fuse or damaging the asset)
Capacity Optimizer	The Capacity Optimizer is responsible for analyzing and optimizing energy usage.
Charging Station	The Charging Station is the physical system where an EV can be charged.
CSO	Charging Station Operator. It is the party that operates a network of charging stations and has contracts with EMSPs to allow their customers to use the charging facilities.
Consumption capacity	Refers to energy flowing from the power grid to a Flexibility Resource.
DSO	Distribution System Operator. The DSO manages the distribution network and has the interest of not overloading the (local) grid.
EMSP	E-Mobility Service Provider. The eMSP is defined as the party that pays for the electricity with which the EV is charged and has a contract with the EV user.
EV	Electric Vehicle
Flexibility Provider	A party that controls flexibility resources and in this way provides flexibility to other parties. Example flexibility providers are Energy Management System operators and Charging Station Operators.
Flexibility Resource	A device that produces or consumes electric energy, such as Charging Stations, batteries or solar panels.
Generation capacity	Refers to energy flowing from a Flexibility Resource to the power grid.
Offline situation	There is no communication possible between 2 systems (e.g. Flexibility Provider and Capacity Provider).
Requirement	Provision that conveys criteria to be fulfilled. ISO/IEC Guide 2:2004, 7.5
Standard Time	Representation of dates and times according to the ISO8601 standard
State Of Charge	The level of charge of an electric battery relative to its capacity. (0% = empty; 100% = full)
String	Case Sensitive String. Only printable ASCII allowed. All strings in messages and enumerations are case sensitive, unless explicitly stated otherwise.
Use case	For complex systems, the use case methodology supports a common understanding of functionalities, actors and processes across different technical committees or even different organizations. Developed as software engineering tool, the methodology can be used to support the development of standards as well as in the analysis of requirements in relation to new or existing standards. Generally, it provides the description of a system's behavior as it responds to a request that originates from outside that system, i.e. the term charging scenario is used simultaneously to the term use case within this document.
Utility	An energy company, such as DSO, TSO, BRP or supplier. Please note that this term may have different meaning per country. Different countries may have these different roles separated in a different way.

1.5. Abbreviations

Table 3. Abbreviations

Abbreviation	Description
BEV	Battery Electric Vehicle
BRP	Balance Responsible Party
CO	Capacity Optimizer
CSMS	Charging Station Management System
CSL	Comma Separated List
CS	Charging Station
CSO	Charging Station Operator
DSO	Distribution System Operator
EMS	Energy Management System
EV	Electric Vehicle
FP	Flexibility Provider
FR	Flexibility Resource
FTP(S)	File Transport Protocol (Secure)
HTTP(S)	HyperText Transport Protocol (Secure)

Abbreviation	Description
NTP	Network Time Protocol
PDU	Protocol Data Unit
SC	Smart Charging
PV	Photovoltaic effect
TLS	Transport Layer Security
TSO	Transmission System Operator
URL	Uniform Resource Locator
UTC	Coordinated Universal Time
WAN	Wide Area Network.

1.6. Actors

This section is informative.

In OSCP, system actors are covering functions or devices.

Table 4. Actors

Actor name	Abbreviation	Actor type	Actor description
Capacity Optimizer	CO	Actor	The Capacity Optimizer is responsible for analyzing and optimizing energy usage.
Flexibility Provider	FP	Actor	A party that controls flexibility resources and in this way provides flexibility to other parties. Example flexibility providers are Energy Management System operators and Charging Station Operators.
Flexibility Resource (or asset)	FR	Device	A device that produces or consumes electric energy, such as Charging Stations, PV panels, (stationary) batteries, etc.
Capacity Provider	CP	Actor	Provides capacity. Will Often refers to an energy company, such as DSO, TSO, BRP or supplier. Could be (a commercial party providing) an Energy Management System.

1.7. References

Table 5. References

Reference	Description
[IEC62559-2:2015]	Definition of the templates for use cases, actor list and requirements list. https://webstore.iec.ch/publication/22349
[OCPP16]	Open charging station Protocol 1.6 OCA, 2015
[OCPP201]	Open charging station Protocol 2.0.1 OCA, 2020
[SECLAQUSO]	An end-to-end security design for smart EV charging for Enexis and Stichting e-laad, LaQuSo, , 2014
[MONTES2013]	A flexible and privacy friendly ICT architecture for Smart Charging of EV's,Proceedings Cired Conference, paper 0199 Montes Portela, Carlos et al. 2013
[NIST_RISK]	Guide for Conducting Risk Assessments NIST – National Institute of Standards and Technology, 2012
[ISO7498-1]	The model provides a common basis for the coordination of standards development for the purpose of systems interconnection, while allowing existing standards to be placed into perspective within the overall Reference Model. The model identifies areas for developing or improving standards. https://www.iso.org/standard/20269.html
[ISO8601]	"Date and time format" http://www.iso.org/iso/home/standards/iso8601.htm
[RFC2119]	"Key words for use in RFCs to Indicate Requirement Levels". S. Bradner. March 1997. http://www.ietf.org/rfc/rfc2119.txt 7498-1
[RFC3339]	"Date and time format" https://www.ietf.org/rfc/rfc3339.txt
[URL]	"URL spec" http://www.w3.org/Addressing/URL/uri-spec.html

1.8. Generic Requirements

This section is normative.

The generic requirements build the basis for defining the use case elements described in the Functional Blocks.

Table 6. Generic requirements

ID	Precondition	Requirement definition
FR.01	Upon receipt of a message from actor X	The receiver of that message SHALL give a response, before sending any other message to actor X.

1.9. Time format requirements

1.9.1. All date-times include a time zone offset

The notation of `date-time` in this protocol are specified in [JSON Schema's](http://json-schema.org/draft/2019-09/json-schema-validation.html#rfc.section.7.3.1) section 7.3.1, which references [RFC3339](https://tools.ietf.org/html/rfc3339#section-5.6) Section 5.6. This defines that all `date-time` **MUST** include a timezone, or it will not pass message validation. Either `Z` for UTC or a numeric offset in the form `+02:00` is acceptable.

This disambiguates any timestamp for the receiving party.

It is recommended, though not required, to convert all datetimes to the UTC time zone before message serialization, but this is not neccesary for unambiguous communication.

2. Architecture and Topology

This section is informative.

2.1. Introduction

The OSCP 2.0 specification is based on the OSCP 1.0 specification, but the terminology and architecture are thoroughly revised. OSCP 1.0 uses terminology such as DSO, CSO, CSP and EV. This imposes a specific domain for OSCP, while its usage can be broader. This section describes the underlying architectural principles on which the interface specification is designed.

2.2. Domain Model

To understand the terminology in the OSCP 2.0 specification, it is important to understand the starting point of this specification. The OSCP specification uses the term **Flexibility Resource**, which is a physical device that can consume or generate energy in a controlled, flexible way. Examples of Flexibility Resources are Electric Vehicles, batteries and heat pumps. Flexibility Resources can be flexible with respect to the time and/or amount they consume or generate energy.

Flexibility Resources are managed by a **Flexibility Provider**. A Flexibility Provider controls a set of Flexibility Resources. It can demand the Flexibility Resources to generate or consume energy. The mechanism for the Flexibility Provider to control the Flexibility Resources is out of scope of OSCP. Examples of Flexibility Providers are charging station operators or battery operators. The mechanism to control the Flexibility Resource is not specified in OSCP.

The Flexibility Providers can receive boundaries for energy consumption or generation from the **Capacity Provider**. A Capacity Provider manages and measures a certain network area and can impose the boundaries to the Flexibility Providers in that area. The Capacity providers do not address the individual Flexibility Resources directly; it is up to the Flexibility Provider to manage his Flexibility Resources in the most economical way to comply to the boundaries of the Capacity Provider.

Figure 1. Overview interrelationships

Examples of a Capacity Provider are a DSO that is responsible for the correct operation of a certain network segment, and an Energy Management System that is responsible to maintain request and demand of energy within the grid connection capacity.

In OSCP we also define the role of a **Capacity Optimizer**. The Capacity Optimizer can support the Flexibility Provider by providing the optimum way to manage the Flexibility Resources. In practice the Capacity Optimizer might have additional data sources such as weather forecasts and historical energy tariffs which can improve the decisions that the Flexibility Provider must take. Capacity Optimizers can be companies that provide optimization services. However, a Capacity Provider or Flexibility Provider can also determine an optimum by themselves.

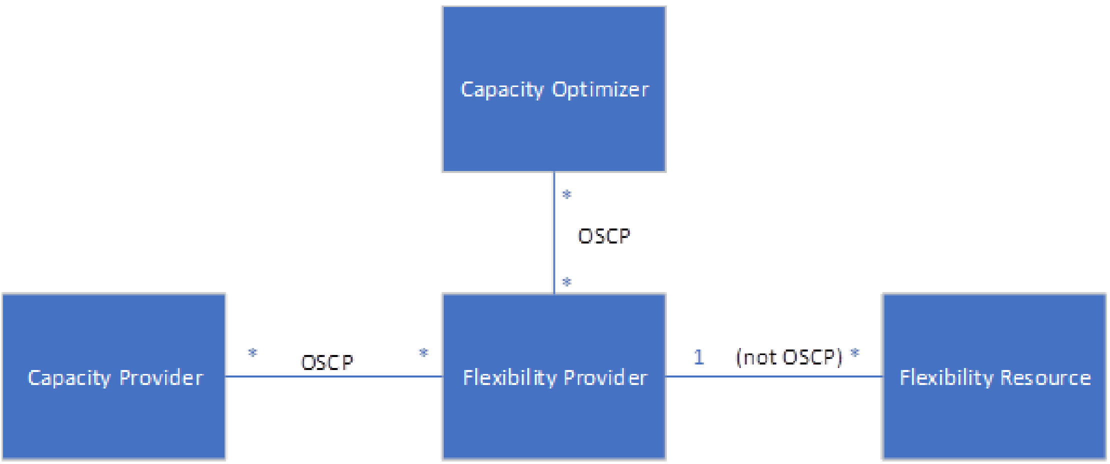


Figure 2. Overview interrelationships with Capacity Optimizer

2.3. Capacity Forecast

A Capacity Provider can submit **Capacity Forecasts** to a Flexibility Provider. A Capacity Forecast is a series of time intervals, and per time interval maximum five so-called capacity types. It SHOULD be noted that the Optimum does not represent a bandwidth, which is the case with the other Capacity Forecasts, but a value. The following types of capacity forecasts can be identified.

- The **Consumption Capacity** (CC) specifies for the given time interval the maximum total amount of capacity that the group of Flexibility Resources can consume. The Consumption Capacity is defined as a positive value. The Flexibility Provider SHALL take measurements not to exceed the Consumption Capacity for the aggregated consumption of the group of Flexibility Resources.
- The **Generation Capacity** (GC) specifies for the given time interval the maximum total amount of capacity that the group of Flexibility Resources can generate. The Generation Capacity is defined as a negative value. This capacity is for example useful in stationary battery discharge use cases.
- The **Fallback Consumption Capacity** (FCC) specifies the maximum total amount of capacity that can be consumed in island mode, i.e. when the Flexibility Provider and the Capacity Provider have lost connection (Offline). The Fallback Consumption Capacity is defined as a positive value.
- The **Fallback Generation Capacity** (FGC) specifies the maximum amount of capacity that can be produced in island mode, i.e. when the Flexibility Provider and the Capacity Provider have lost connection (Offline). The Fallback Generation Capacity is defined as a negative value.
- The **Optimum** (O) specifies for the given time interval the Optimum amount to be generated or consumed. The interpretation of the Optimum is not specified by OSCP; it shall be based on a mutual agreement between the Capacity Provider and the Flexibility Provider.

Table 7. An example layout for a capacity forecast is given:

Time interval	CC	GC	FCC	FGC	O
T1	32	-10	8	-5	24
T2	16	-10	8	-5	15
T3	32	-10	8	-5	-4
T4	16	-10	8	-5	15

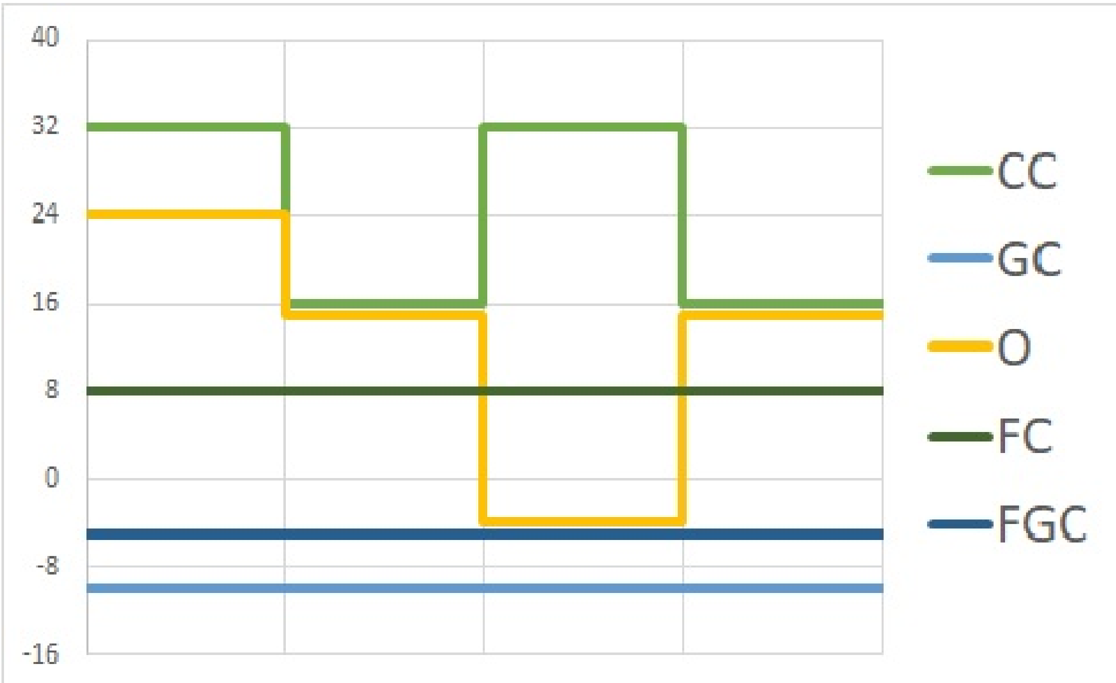


Figure 3. Plot of the example capacity forecast

This example illustrates for time interval T1 that:

- The Flexibility Provider can consume between 0 and 32 (for instance current or power)
- The Flexibility Provider can generate between -10 and 0 (for instance current or power)
- The Flexibility Provider has received an optimum of 24, which means that he might have an incentive to strive for a consumption of 24.

2.3.1. Request for adjustment of capacity

If the Flexibility Provider needs more capacity than assigned, it can request additional capacity to the Capacity Provider. Or, if the Flexibility Provider needs less capacity than assigned, it can request for less capacity to the Capacity Provider. OSCP supports both situation by means of a AdjustGroupCapacityForecast message, which is not available if the Capacity Provider role is fulfilled by a DSO or TSO, unless this is explicitly mentioned. Both are non-discriminatory parties, which are not allowed to distinguish between Flexibility Provider.

2.4. Capacity Optimizer

The Capacity Optimizer has the responsibility to add the Optimum to the time interval. The Capacity Optimizer uses the capacity forecast and can use external information, not in scope of OSCP, to determine the Optimum. Examples of external information are energy prices, weather forecast and energy usage prognoses. Capacity Optimizers can be part of specialized companies, which is the reason to model them as a separate role.

2.5. Metering

The Flexibility Provider has the possibility to send metering information to the Capacity Provider and to the Capacity Optimizer. There are two types of measurements:

- Accumulated energy measurements of an aggregated area (**group**) of Flexibility Resources
- Flexibility Resource (**asset**) specific measurements

2.5.1. Group Measurements

The metering data of an aggregated area (group) provides information over the total consumption or production of **energy** (in (k)Wh) by the Flexibility Provider. The Capacity Provider can validate with this information if and how the Flexibility Provider has operated within the given capacities. Additionally, the information can be used as input for the Capacity Forecast distribution by the Capacity Provider over Flexibility Providers.

2.5.2. Asset Measurements

This metering data provides information on the usage per Flexibility Resource. Both energy and instantaneous measurements are supported. These measurements are especially relevant for the Capacity Optimizer. The Capacity Optimizer can use this information in its Capacity Forecast and Optimization algorithms.

2.6. Connection

The following sections describes connectivity related challenges like registering endpoints, connecting (handshaking) and handling connection issues.

2.6.1. Registration

Registering endpoints is needed in order to make sure the received messages on these endpoints actually come from the designated party. For instance as a Flexibility Provider I want to make sure the Capacity Forecast (that can potentially have a negative influence on the experience of my customers) actually comes from my trusted Capacity Provider.

To register one party to another, every party must create a unique token to use for authentication. This token will have to be sent from one party to the other in a secure way that is outside the scope of this protocol. Every endpoint ought to be registered using a token.

The (one-time) registration of an endpoint MUST be done **prior** to sending handshakes which is described below.

2.6.2. Handshaking

Each OSCP communication between two parties shall start with 'a handshake'. Using this mechanism a party can express their preferences to the other party. For instance a Flexibility Provider instantiates a handshake with a Capacity Optimizer to define the interval in which it expects the heartbeats.

A handshake should be acknowledged (expressing preferences as well) and after the acknowledgement is replied with an HTTP-204 the handshake sequence is completed. From this moment on the connection is established (*Online*), and the sending of other

OSCP message is permitted. When handshaking is not (yet) completed any other message SHOULD either be rejected or ignored.

A handshake MUST be instantiated upon startup. Handshakes can be initiated at any time.

Every party can initiate a handshake, but it is commonly initiated by the party benefiting most from the connection. For instance a grid operator (as Capacity Provider) carries the final responsibility for keeping the grid stable, it therefore has a bigger interest in a reliable OSCP connection with its Flexibility Providers than vice versa.

2.6.3. Offline detection

For every party it is important to have a solid connection with the other party. For instance its important for the Flexibility Provider to know whether the received capacity forecast is still actual; when the Capacity Provider is not reachable, the Flexibility Provider has to assume that the capacity forecast is not actual, and the fallback capacity will have to be followed. Offline detection can be done by the heartbeat mechanism (recommended) or by discovering the lack of new capacity forecasts or meter values.

When a party is classified as *Offline* it can be assumed the party is (temporary) unavailable.

The higher the heartbeat frequency, the more adequate the offline detection is.

2.6.4. Heartbeats

In order to guarantee the *Online* mode, heartbeat messages are sent between parties. Heartbeats are primarily used to detect unwanted deviations in the connection.

A heartbeat message contains a datetime field. If no heartbeat messages have been received before this time, the other party is *Offline*. As long as heartbeat messages are transferred the datetime field will be updated with every heartbeat message.

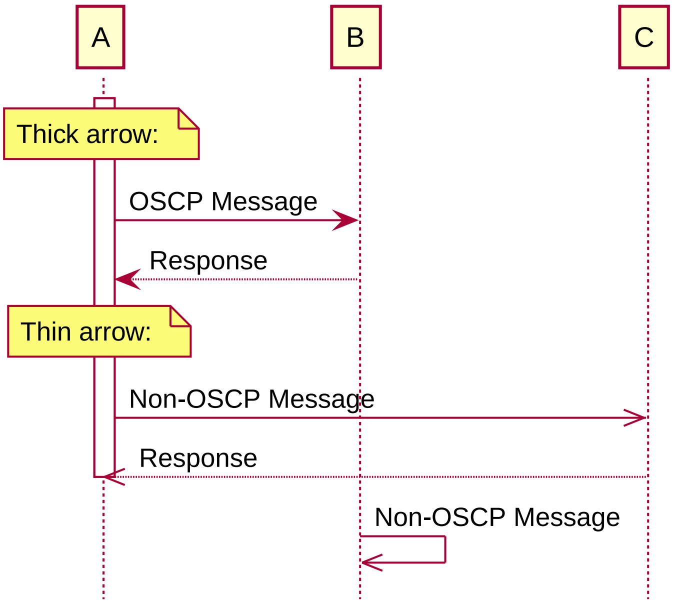
Configuring the interval in which heartbeats should be sent is part of the handshaking mechanism. The message frequency SHOULD be much higher than that of any other message in order to make sense.

3. Use Cases

In this chapter several use cases are described. The use cases are divided into four categories which are: Capacity Distribution, Measurements, Registration and Fallback & Errors.

Unless otherwise specified the following applies in this chapter:

- In the following Sequence Diagrams the thin arrowheads are defined as out of scope of the OSCP protocol.



- All use cases and requirements assume normal communication between parties (*Online*).

3.1. Connectivity

This section describes how a Flexibility Provider registers itself to a Capacity Provider to establish a connection.

3.1.1. Flexibility Provider registers its Capacity capabilities to a Capacity Provider

3.1.1.1. Use case description

No.	Type	Description
1	**Name**	Flexibility Provider registers itself to a Capacity Provider.
2	**ID**	5
3	**Objectives**	Making sure the received messages are actually from the the designated party.
4	**Description**	In order for the FP to be able to serve the CP, authentication should be configured at both sides.
	Actors	FP, CP
	Scenario description	**1.** The FP generates a token (TOKEN_A). This token along with the base endpoint of FP is sent to the CP in some secure way that is outside the scope of OSCP. **2.** The CP generates a token (TOKEN_B) that is to be used for authenticating FP at CP. The CP sends the token along with the base endpoints per OSCP version that it supports to the FP via a Register message. CP uses TOKEN_A to authenticate at FP. **3.** FP replies with HTTP 204 if it supports one of the presented OSCP versions and chooses the OSCP version that it wants to use from the list provided by CP and it generates a new token (TOKEN_C) that is to be used by CP from now on. **4.** CP replies with HTTP 204.
5	**Prerequisites**	FP and CP have never communicated with each other.

6	**Postcondition(s)**	**Successful postcondition:** The registration is completed. CP can continue with Capacity Provider handshakes with Flexibility Provider. **Failure postcondition:** - FP does not support any of the OSCP versions from CP. FP returns HTTP 501 (Not Implemented) - The registration is completed but the CP is still able to authenticate at FP using the initial token (TOKEN_A).

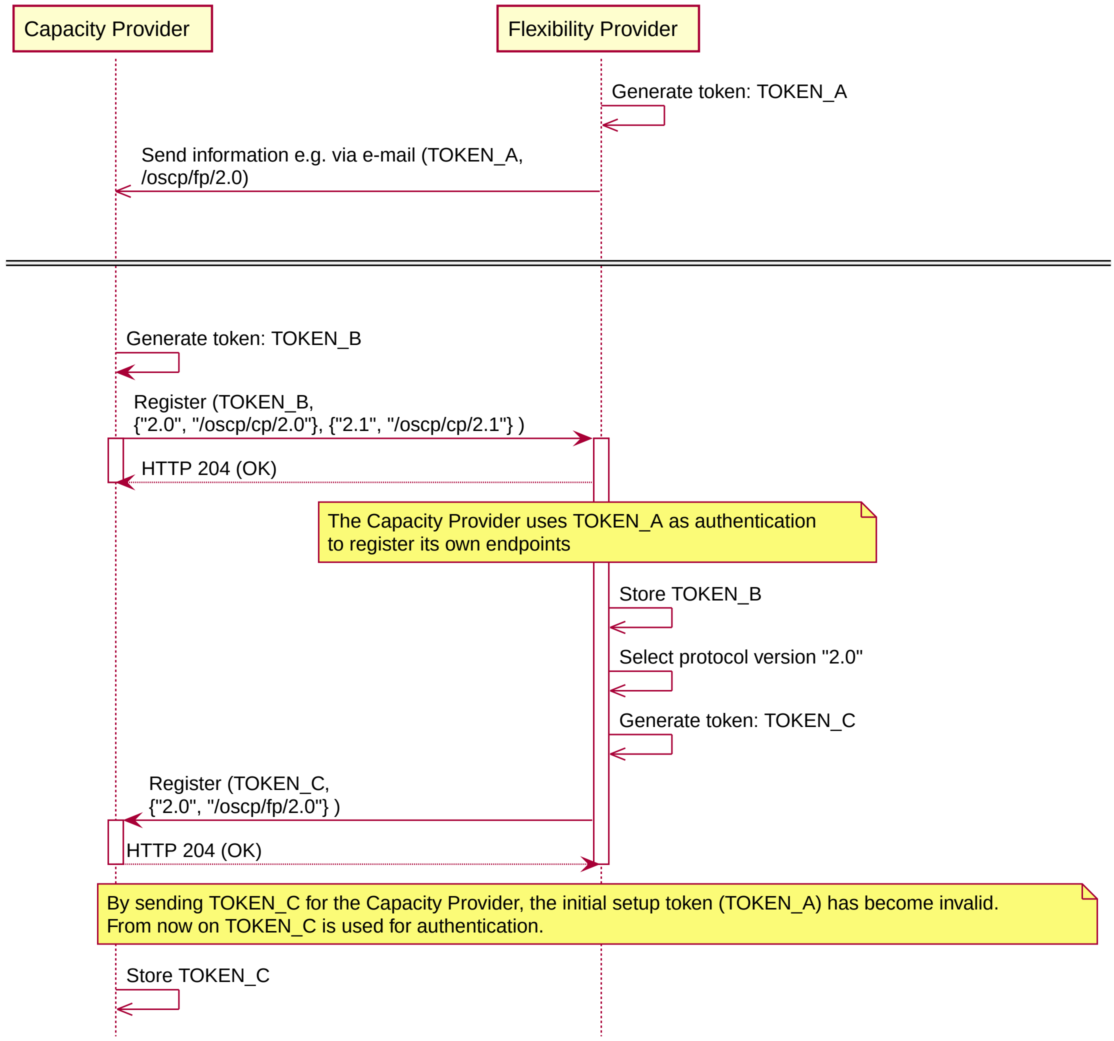


Figure 4. Sequence Diagram: Flexibility Provider registers itself to a Capacity Provider

7	**Error Handling**	n/a
8	**Remarks**	The registration MUST be completed prior to the sending of any other message. This should be a only once, but the registration process can be repeated in case a new token needs to be registered.
9	**Example**	n/a

3.1.2. Capacity Provider handshakes with Flexibility Provider

3.1.2.1. Use case description

No.	Type	Description
1	**Name**	Capacity Provider handshakes with Flexibility Provider.
2	**ID**	6
3	**Objectives**	Announcing preferences between parties, prior to any other OSCP communication (besides registration).
4	**Description**	The CP and FP should come to an agreement on their preferences like heartbeat intervals. The handshaking mechanism allows for this.
	Actors	FP, CP

	Scenario description	**1.** The CP sends a Handshake message to the FP. **2.** The FP accepts the handshake by replying with a HTTP 204 and sending a HandshakeAcknowledge message to the CP. **3.** The CP accepts the handshake acknowledge by replying with a HTTP 204.
5	**Prerequisites**	FP is registered at the CP.
6	**Postcondition(s)**	**Successful postcondition:** A completed handshake sequence; both parties have expressed their own preference and have accepted the preferences of the other. From this moment on the connection is established (*Online*) and the sending of other OSCP message is permitted. **Failure postcondition:** - The FP does not accept the CPs preference. It therefore replies to the Handshake message with a HTTP 400 (bad request). - The CP does not accept the FPs preference. It therefore replies to the HandshakeAcknowledge message with a HTTP 400 (bad request).

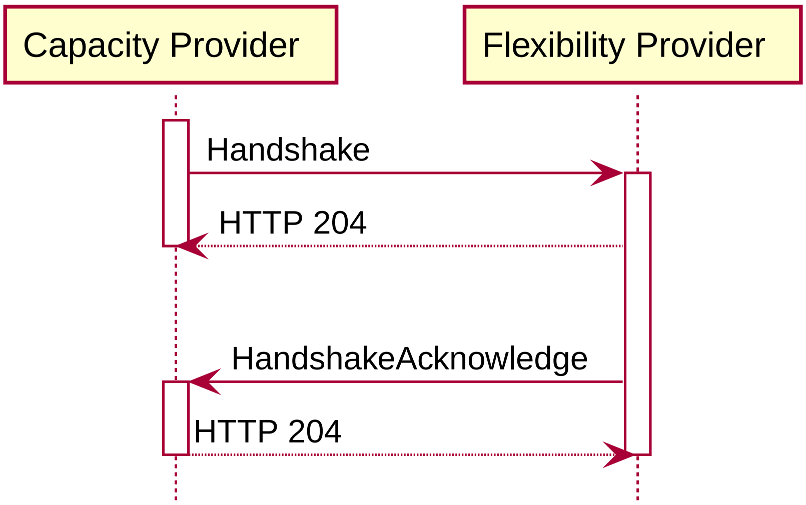


Figure 5. Sequence Diagram: Capacity Provider handshakes with Flexibility Provider

7	**Error Handling**	As long as handshaking is not completed any other message SHOULD either be ignored (not replied to) or rejected, e.g. replied to with a HTTP 403 (forbidden).
8	**Remarks**	Handshaking is mainly used for recovering from *Offline* mode, but is also useful in case preferences change while *Online*.
9	**Example**	n/a

3.2. Distribute Capacity

3.2.1. Capacity Provider distributes Capacities to Flexibility Provider(s)

No.	Type	Description
1	**Name**	Capacity Provider distributes Capacities to one or more Flexibility Provider(s)
2	**ID**	1
3	**Objectives**	The CP distributes various types of capacities to one or more FP(s) to enable them to adjust the load of their FRs, if needed.
4	**Description**	A CP distributes Capacities to one or more FP(s). The available Capacity types are: *Consumption*, *Generation*, *Fallback Consumption*, *Fallback Generation* and *Optimum*. For more information see Capacity Forecast.
	Actors	CP, FP, FR
	Scenario description	**1.** The CP receives measurements from a metering device in, for example, a transformer station (outside of the OSCP protocol). **2.** Based on these measurements, and optionally other input parameters, a forecast of the Capacities is calculated. (outside of the OSCP protocol). **3.** The Capacities are distributes to each individual FP using the UpdateGroupCapacityForecast message to inform the FP about the Capacities that it MUST use. **4.** Based on this forecast the FP can tell its FRs what their Capacities are for the next period of time. **5.** Meanwhile the FP receives metering data from Flexibility Resource(s). **6.** The FP reports aggregated measurements of all FRs to the CP using the UpdateGroupMeasurements message.

5	**Prerequisites**	n/a
6	**Postcondition(s)**	**Successful postcondition:** - All relevant FP(s) have received a forecast of Capacities and have adjusted their FR(s) (if needed). **Failure postcondition:** - The forecast of Capacities has not arrived at the relevant FP(s). - The FR(s) are not adjusted to the Capacities when needed.

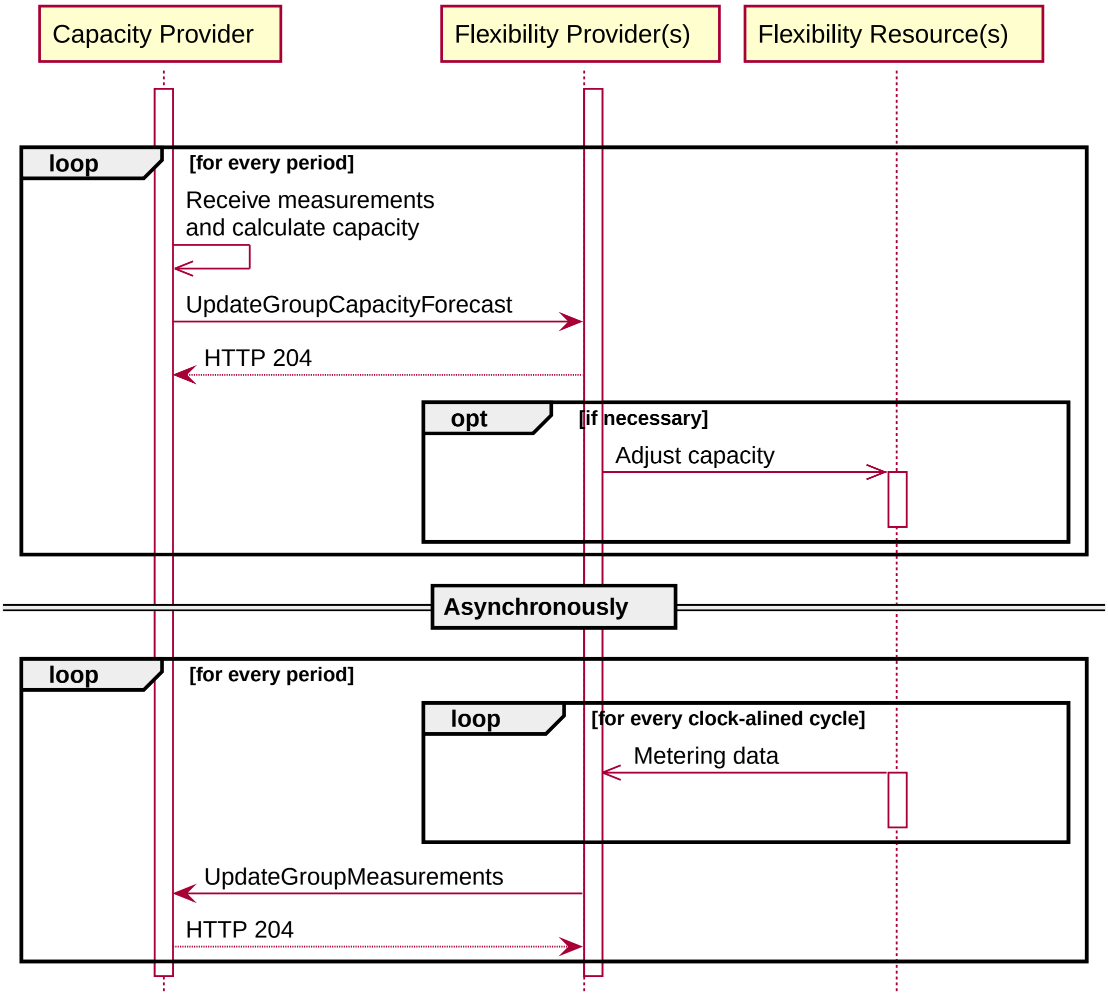


Figure 6. Sequence Diagram: Capacity Provider distributes Consumption Capacity to Flexibility Provider(s)

7	**Error Handling**	n/a
8	**Remarks**	To avoid unfair competition, the distribution during step 3 should be Fair, Reasonable and Non-Discriminatory (FRAND). Whether this is done using contracts or for example based on historical usage is up to the regulation / agreement in the market.
9	**Example 1**	**Distribution of Consumption Capacity**: A grid operator (DSO) takes the role of CP. A charging station operator (CSO) takes the role of FP, with its charging stations (CS) serving as FRs. The DSO measures a transformer station and calculates the Capacities available for the FP. It updates its forecasts of the next 24 hours of the *Consumption* Capacity every 15 minutes and sends this to the FP with the UpdateGroupCapacityForecast message. The CSO reports every 15 minutes with an UpdateGroupMeasurements message.
	Example 2	**Distribution of Consumption and Generation Capacity**: A grid operator (DSO) takes the role of CP. A charging station operator (CSO) takes the role of FP, with its charging stations (CS) serving as FRs. The DSO measures a transformer station and calculates the Capacities available for the FP. It updates its forecasts of the *Consumption* and *Generation* Capacity every 15 minutes and sends this to the FP with the UpdateGroupCapacityForecast message. The CSO reports every 15 minutes with an UpdateGroupMeasurements message.

	Example 3	**Distribution of Consumption and Generation Capacity and Optimum**: An energy management system (EMS) serves as both CP and CO. It measures local renewable power *Generation* (e.g. solar panels) and *Consumption* in an office building. The FP manages energy storage and Charging Stations on the building's premises. Every 15 minutes the CP updates the Capacity forecast for *Consumption* and *Generation* to the FP, including an *Optimum* that maximizes the usage of locally generated renewable energy. The *Optimum* is calculated with weather forecasts by the EMS and results in a trend line during the day, which the FP tries to follow carefully with its FRs (the stationary battery and the Charging Stations). The FP updates the CP with meter values, using the UpdateAssetMeasurements message every 15 minutes to report on actual usage so the CP can better optimize the Capacity for renewable usage and to evaluate the FP's attempts to use the *Optimum* throughout the day.
	Example 4	**Distribution of Consumption, Generation, Fallback Consumption and Fallback Generation Capacity**: An energy management system (EMS) serves as CP and measures local power *Consumption* in an office building. The FP manages energy storage and Charging Stations on the building's premises. Every 15 minutes the CP updates the Capacity forecast available for *Consumption*, *Generation*, *Fallback Consumption* and *Fallback Generation* to the FP. The *Fallback Consumption* and *Fallback Generation* Capacity SHALL be used when an offline situation occurs. The FP updates the CP with aggregated meter values, using the UpdateGroupMeasurements message every 15 minutes to report on actual usage.
	Example 5	**Distribution of real time Capacity**: A grid operator (DSO) serves as CP and measures the local load on a transformer station. The grid operator can send real time Capacities based on these measurements using the UpdateGroupCapacityForecast message. The FP shall use the new Capacity to change the Capacity of the FR(s) they control.
	Example 6	**Distribution of real time Capacity**: An Energy Management System (EMS) Operator serves as CP and measures the local power consumption in a building. The EMS (Operator) can send real time Capacities based on these measurements using the UpdateGroupCapacityForecast message. The FP can choose to use the new Capacity to change the *Consumption* Capacity of the FRs it controls.

3.2.1.1. Requirements

ID	Requirements
FR.01.01	The FP SHALL send aggregated measurements to the CP. This MAY be done with the same interval as that with which the CP sends Capacity forecasts.
FR.01.02	When the CP can no longer guarantee the validity of the Capacity in the UpdateGroupCapacityForecast message (e.g. after connection loss of its meters), the CP SHALL update the Consumption (and Generation) Capacity in a new UpdateGroupCapacityForecast message to a value that can be guaranteed to be used by the FP in a safe manner.
FR.01.03	The *Fallback consumption* and *Fallback generation* Capacity SHALL be used by the FP when an offline situation occurs.
FR.01.04	The FP SHALL use the Capacity from the last received UpdateGroupCapacityForecast message as the new Capacity for its FRs.
FR.01.05	The Consumption Capacity MUST be greater than or equal to 0.
FR.01.06	The Generation Capacity MUST be less than or equal to 0.
FR.01.07	The Fallback Consumption Capacity MUST be less than or equal to the Consumption Capacity.
FR.01.08	The Fallback Generation Capacity MUST be less than or equal to the Generation Capacity.
FR.01.09	The Optimum MUST be less than or equal to the Consumption Capacity and MUST be greater than or equal to the Generation Capacity.

3.2.2. Capacity Optimizer distributes an Optimum

3.2.2.1. Use case description

No.	Type	Description
1	**Name**	Capacity Optimizer distributes an Optimum based on the Capacity from the Capacity Provider.

2	**ID**	2
3	**Objectives**	The CO calculates an Optimum based on the Capacity from the CP and distributes it to the FP.
4	**Description**	To prevent local grid problems, the CP can measure the load on the local electricity grid and based on the Capacity of the grid components, the CP can distribute the Capacity. Within these capacities, the actual usage is optimized by a CO.
	Actors	CP, FP, CO, FR
	Scenario description	**1.** The CP receives measurements from a metering device in, for example, a transformer station (outside of the OSCP protocol). **2.** Based on these measurements, and optionally other input parameters, a forecast of the Capacities is calculated. (outside of the OSCP protocol). **3.** The Capacities are distributes to each individual FP using the UpdateGroupCapacityForecast message to inform the FP about the Capacities that it MUST use. **4.** The FP distributes the UpdateGroupCapacityForecast message to the CO. **5.** After the CO has calculated the Optimum, it is sent to the FP using the UpdateGroupCapacityForecast message. **6.** Based on this Optimum, the FP can tell its FRs what their Capacity for the next period of time is. **7.** Meanwhile the FP receives metering data from its FR(s). **8.** The FP reports aggregated measurements of all FRs to the CP using the UpdateGroupMeasurements message. **9.** The FP reports meter data to the CO using the UpdateAssetMeasurements message.
5	**Prerequisites**	The FP has received a UpdateGroupCapacityForecast from the CP.
6	**Postcondition(s)**	**Successful postcondition:** The FP has received an Optimum and has, depending on the Capacity, changed the used Capacity on one or more FRs. **Failure postcondition:** - The forecast of Capacity has not arrived at the FP. - The forecast of Capacity has not arrived at the CO. - The Optimum has not arrived at the FP.

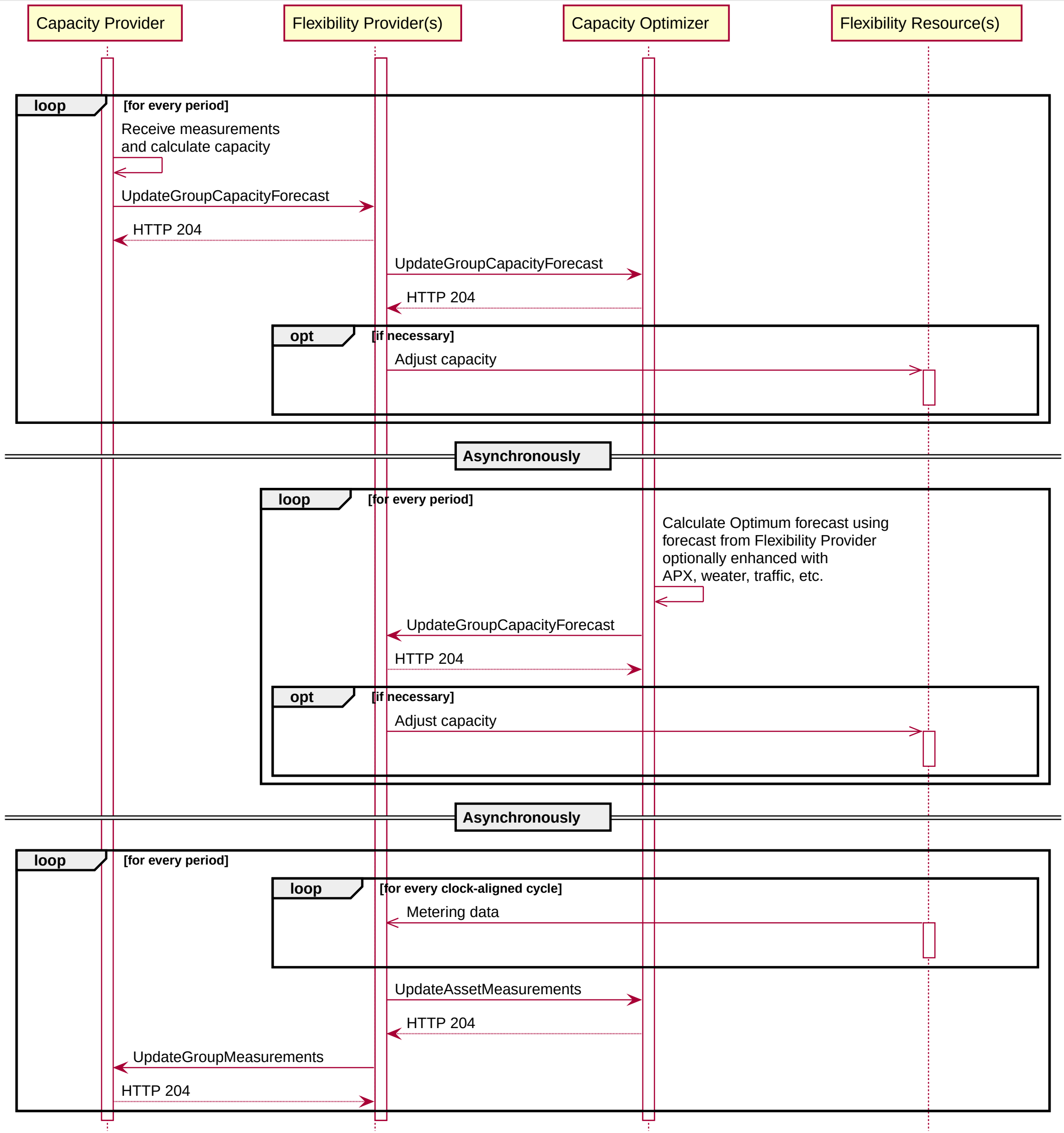


Figure 7. Sequence Diagram: Capacity Optimizer optimizes Capacity distributed by Capacity Provider

7	**Error Handling**	n/a
8	**Remarks**	To avoid unfair competition, the distribution during step 3 should be Fair, Reasonable and Non-Discriminatory (FRAND). Whether this is done using contracts or for example based on historical usage is up to the regulation / agreement in the market.
9	**Example 1**	A grid operator serves as CP and measures the local load on a transformer station powering a campus. On this campus, several power generation (PV) and storage (stationary batteries) devices are measured by a third party, serving as CO. The stationary battery and several EV chargers are managed by the FP. For determining the Capacity that can be allocated to this group of devices, the FP receives messages from the CP every 15 minutes. In case the CP stops sending these messages for undisclosed reasons, the FP resorts to the fallback capacities in the latest received UpdateGroupCapacityForecast message. To optimize the control of the FRs for maximum usage of the local renewable power generation, the FP forwards the received UpdateGroupCapacityForecast messages to the CO. The CO sends an UpdateGroupCapacityForecast message to the FP every 15 minutes, now including an *Optimum* forecast. The forecast is calculated with weather forecasts and the CO's measurements on the PV system and batteries. The FP uses the *Optimum* to control the stationary batteries and EV chargers and reports to the grid operator by sending it UpdateGroupMeasurements messages and to the CO by sending it UpdateAssetMeasurements messages every 15 minutes.

	Example 2	A grid operator (DSO) takes the role of CP and a Charging Station Operator (CSO) takes the role of FP. The DSO receives measurements from the transformer station. Based on these measurements every period of 15 minutes a 24 hour rolling forecast of the available group *Consumption* Capacity is made. This forecast is forwarded to the CSO using the UpdateGroupCapacityForecast message. The CSO redirects the message to the CO and optionally adapts the consumption of its Charging Stations (FR's). Subsequently, the EVs at the Charging Stations can be charged, be it throttled, depending on the Capacity. Meanwhile, once every 15 minutes, the CO has calculated the groups' *Optimum* forecast using all sorts of inputs including the *Consumption* forecast from the CSO. The CO sends this forecast to the CSO. Based on this forecast the CSO can adapt the consumption of its Charging Stations as well. Meanwhile, once every 15 minutes, the CSO is assembling metering data from the Charging Stations. The CSO combines all metering data assembled over the last period and wraps it in an UpdateAssetMeasurements message that it sends to the CMO. The same goes for the DSO, but since it has an aggregation interface, the CSO sends an UpdateGroupMeasurements message to it.

3.2.2.2. Requirements

ID	Requirements
FR.02.01	The FP SHALL send aggregated measurements to the CP. This MAY be done with the same interval as that with which the CP sends Capacity forecasts.
FR.02.02	The CP can alter the Capacity without any limits at any time using the UpdateGroupCapacityForecast message. For example, a CP can use this to guarantee grid stability when it has lost insight of the grid.
FR.02.03	When the FP no longer receives the UpdateGroupCapacityForecast messages from the CP, the FP SHALL use the Fallback Capacity from the last received UpdateGroupCapacityForecast message as the new Capacity for its FRs.
FR.02.04	The Consumption Capacity MUST be greater than or equal to 0.
FR.02.05	The Generation Capacity MUST be less than or equal to 0.
FR.02.06	The Fallback Consumption Capacity MUST be less than or equal to the Consumption Capacity.
FR.02.07	The Fallback Generation Capacity MUST be greater than or equal to the Generation Capacity.

3.2.3. Flexibility Provider request additional Capacity

3.2.3.1. Use case description

No.	Type	Description
1	**Name**	Flexibility Provider request additional Capacity.
2	**ID**	3
3	**Objectives**	The FP request additional Capacity from the CP based on its updated Capacity needs.
4	**Description**	The Capacity that is requested by the FR(s) does not comply with the distributed Capacity. Therefore the FP request additional Capacity.
	Actors	CP, FP
	Scenario description	**1.** The CP distributes the UpdateGroupCapacityForecast message to the FP. **2.** The distributed Capacity does not match the demand of Capacity from the FP. **3.** The FP request additional Capacity from the CP using the AdjustGroupCapacityForecast message.
5	**Prerequisites**	The FP has received a UpdateGroupCapacityForecast from the CP.
6	**Postcondition(s)**	**Successful postcondition:** The CP has received the request from the FP. **Failure postcondition:** The CP did not received the request from the FP.

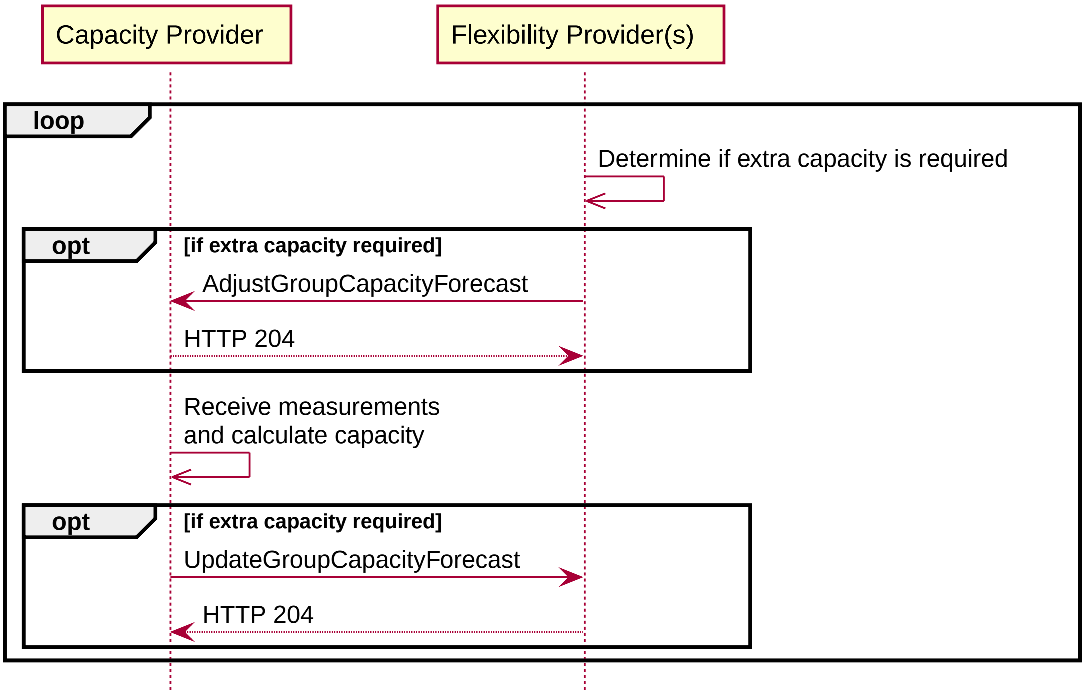


Figure 8. Sequence Diagram: Flexibility Provider request additional Capacity

7	**Error Handling**	n/a
8	**Remarks**	A DSO is a non discriminatory party, it is not allowed to distinguish between FPs. OSCP does not prescribe how to distribute the Capacity between various FPs (Out of scope).
9	**Example**	When a CP provides Capacity to multiple FPs on the same group, the CP chooses to hand out less Capacity than available to the FPs. The FP can then request additional Capacity from the CP in case it is needed. Since some Flexibility Providers can be more flexible than others (e.g. those controlling heat pumps versus those controlling EV chargers), this will allow for further fine-tuning of system-wide flexibility.

3.3. Distribute measurements

3.3.1. Capacity Optimizer enhances optimization based on charging (EV) session information

3.3.1.1. Use case description

No.	Type	Description
1	**Name**	Capacity Optimizer enhances optimization based on charging (EV) session information.
2	**ID**	4
3	**Objectives**	Enhance the forecasted optimum based on the charging (EV) session information.
4	**Description**	The CO serves the FP in distributing the amount of energy over its charge points (FRs) as efficient as possible.
	Actors	FP, FR, CO, EV + Driver
	Scenario description	**1.** The EV Driver starts the process by plugging in the cable and swiping the identification card. **2.** After authorization the EV starts charging complying to the (default) limit **3.** While the charging session is active the FR sends metering data to the FP. **4.** Every period the FP uses the UpdateAssetMeasurements message to inform the CO. **5.** Every period the CO calculates a new Optimum forecast and send it to the FP using the UpdateGroupCapacityForecast message. **6.** If necessary the FP adjusts the charging limit of the FR. **7.** Than the EV adjusts to this charging speed. **8.** After some time the EV Driver ends the charging process by swiping the identification card again and unplugging the cable.
5	**Prerequisites**	Capacity forecasts are distributed from CP to FP.

6	**Postcondition(s)**	**Successful postcondition:** The EV adjust its charging speed when this was requested. **Failure postcondition:** The EV did not adjust its charging speed when this was requested.

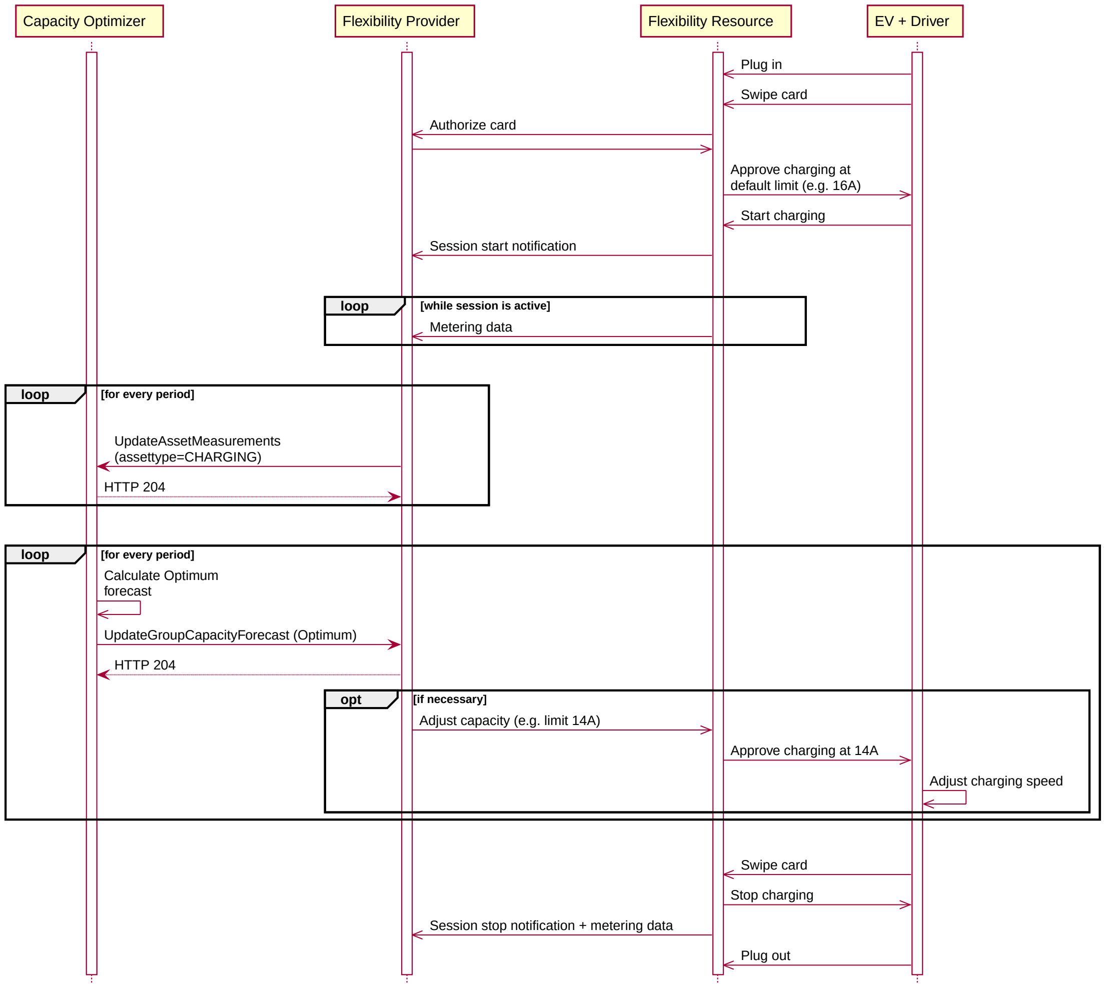


Figure 9. Sequence Diagram: Capacity Optimizer enhances optimization based on charging (EV) session information

7	**Error Handling**	n/a
8	**Remarks**	The Optimum forecast can be improved by informing the CO about the end time and/or required energy of the charging session.
9	**Example**	To improve the optimization of the Capacity, the CO uses EV session information to predict connection times, total energy, etc.

3.4. Fallback and error situations

3.4.1. Flexibility Provider cannot comply to Capacities

3.4.1.1. Use case description

No.	Type	Description
1	**Name**	Flexibility Provider cannot comply to Capacities
2	**ID**	7
3	**Objectives**	Enable the FP to inform the CP that it can possibly not comply to the Capacities.
4	**Description**	If the FP cannot comply to the Capacities (e.g. due to an occurred event). Using this use case, the FP is able to inform the CP.
	Actors	CP, FP, FR

	Scenario description	**1.** The CP sends the available Capacity to each individual FP using the UpdateGroupCapacityForecast message to inform the FP about the Capacity that it may use. **2.** Based on this forecast the FP can tell it's FRs what their Capacity for the next period is. **3.** An event occurs which makes it impossible to comply to the distributed Capacity. **4.** The FP informs the CP using the [groupcapacitycomplianceerror-definition] message.
5	**Prerequisites**	n/a
6	**Postcondition(s)**	**Successful postcondition:** The CP is informed that the FP cannot comply to Capacities. **Failure postcondition:** The CP does not know that the FP cannot comply to Capacities.

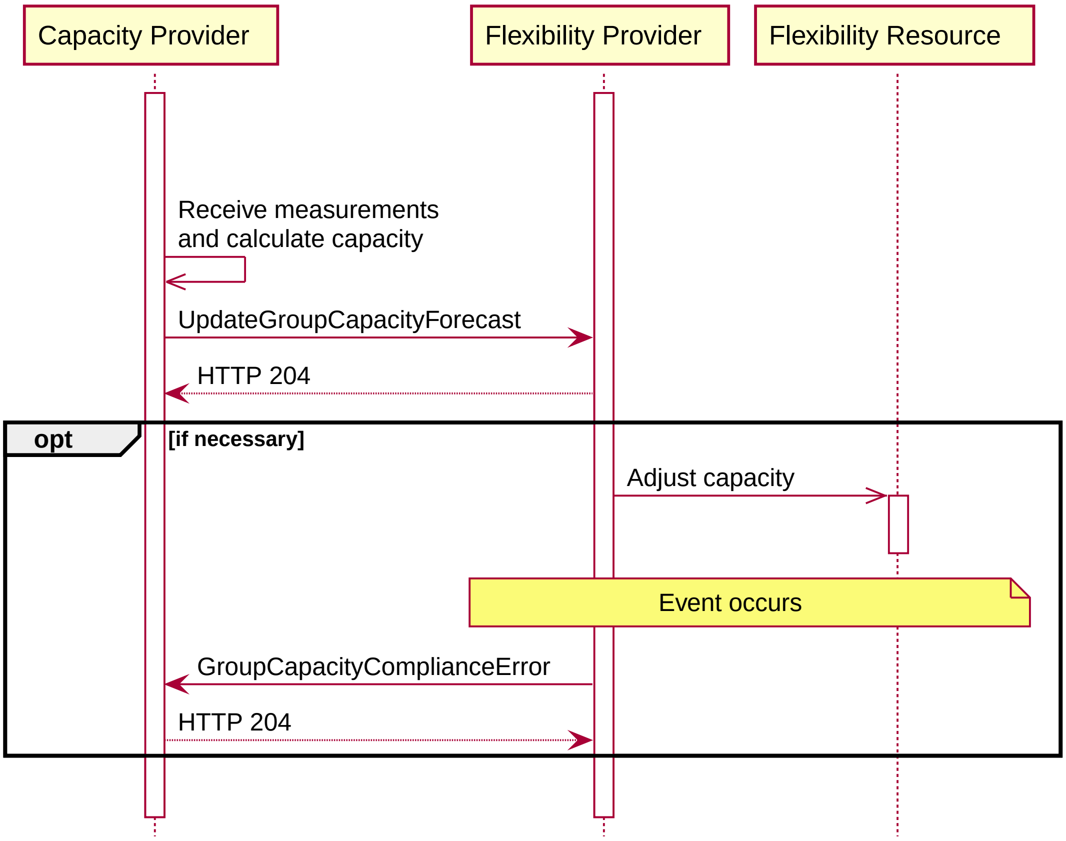


Figure 10. Sequence Diagram: Flexibility Provider reports it cannot comply with the Capacity Provider profile

7	**Error Handling**	n/a
8	**Remarks**	n/a
9	**Example 1**	The communication network used by the FP to control the Flexibility Resources is down. Therefore it is not possible to comply with the Capacities from the CP. The FP uses the [groupcapacitycomplianceerror-definition] message to inform the CP.
	Example 2	If the FP receives an UpdateGroupCapacityForecast message but still cannot comply the FP sends a [groupcapacitycomplianceerror-definition] message. The CP then sends an adapted UpdateGroupCapacityForecast message. The FP determines it can comply to this forecast and does not send the [groupcapacitycomplianceerror-definition] message, therefore the CP can assume that the FP can comply again.

3.4.1.2. Requirements

ID	Requirements
FR.04.01	If the FP *cannot comply* this MUST be mentioned to the CP as soon as possible using the [groupcapacitycomplianceerror-definition] message.
FR.04.02	If the FP has no other option than to exceed the Capacity given by the CP than it can be assumed that the FP *cannot comply*. See requirement above.
FR.04.03	If the FP receives a UpdateGroupCapacityForecast message but still cannot comply this MUST be mentioned again using the [groupcapacitycomplianceerror-definition] message.
FR.04.04	If the FP receives a UpdateGroupCapacityForecast message and does not send the [groupcapacitycomplianceerror-definition] message it is assumed that the FP can comply again.

3.4.2. Detect an offline situation

3.4.2.1. Use case description

No.	Type	Description
1	**Name**	Detect an offline situation
2	**ID**	8
3	**Objectives**	Determine when the other party is offline.
4	**Description**	The heartbeat can be used to detect if the other party is offline. In this use case we use Party A and Party B, because it is valid for all actors.
	Actors	CP, FP, FR, CO
	Scenario description	**1.** Party A sends a heartbeat to Party B with a certain interval. This heartbeat contains a offline_mode_at = datetime, which is a time in the future that is consecutive with each new interval. **2.** The server from Party A breaks down, so Party A cannot send heartbeats to Party B **3.** If Party B does not receive any heartbeat before offline_mode_at = datetime it can conclude that Party A is offline.
6	**Postcondition(s)**	**Successful postcondition:** Party B detected that Party A is offline. **Failure postcondition:** Party B did not detect that Party A is offline. Party B concluded that Party A is offline while this is not true.

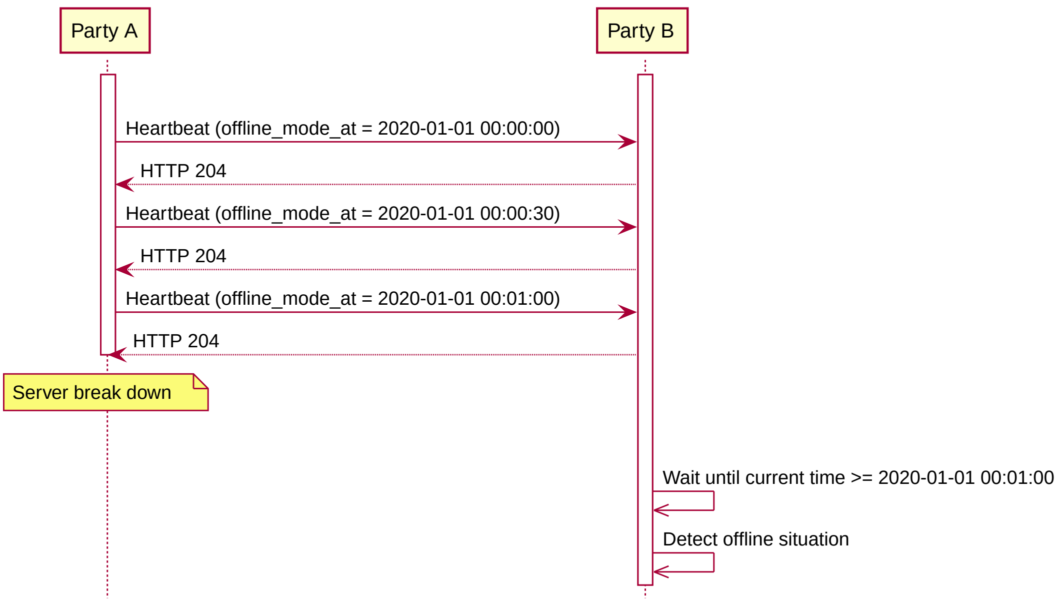


Figure 11. Sequence Diagram: Party B detects that party A is offline

7	**Error Handling**	n/a
8	**Remarks**	n/a
9	**Example**	n/a

3.4.3. Flexibility Provider adapts to a situation where the Capacity Provider is offline

3.4.3.1. Use case description

No.	Type	Description
1	**Name**	Flexibility Provider adapts to a situation where the Capacity Provider is offline.
2	**ID**	9
3	**Objectives**	Guarantee the stability of the grid by using Fallback Capacities that can be used when the communication between the Capacity Provider and FP is lost.
4	**Description**	If an offline situation is detected the FP MUST adapt its FRs to the Fallback Capacities to guarantee grid stability.
	Actors	CP, FP, FR

	Scenario description	**1.** An offline situation occurs which makes it impossible for the CP to send the UpdateGroupCapacityForecast message to the FP. **2.** The FP adapts the FRs to use the Consumption and Generation Fallback Capacities. **3.** The CP is online again and sends a Handshake **4.** The FP acknowledges the Handshake and detects that the situation is restored. **5.** The FP adapts the FRs to use the Consumption and Generation Capacities.
5	**Prerequisites**	The FP detected that the CP is offline, see use case Detect an offline situation.
6	**Postcondition(s)**	**Successful postcondition:** The FRs are adapted to the Consumption and Generation Fallback Capacities. **Failure postcondition:** The FRs are not adapted to Fallback Capacities and still use Consumption and Generation Capacities.

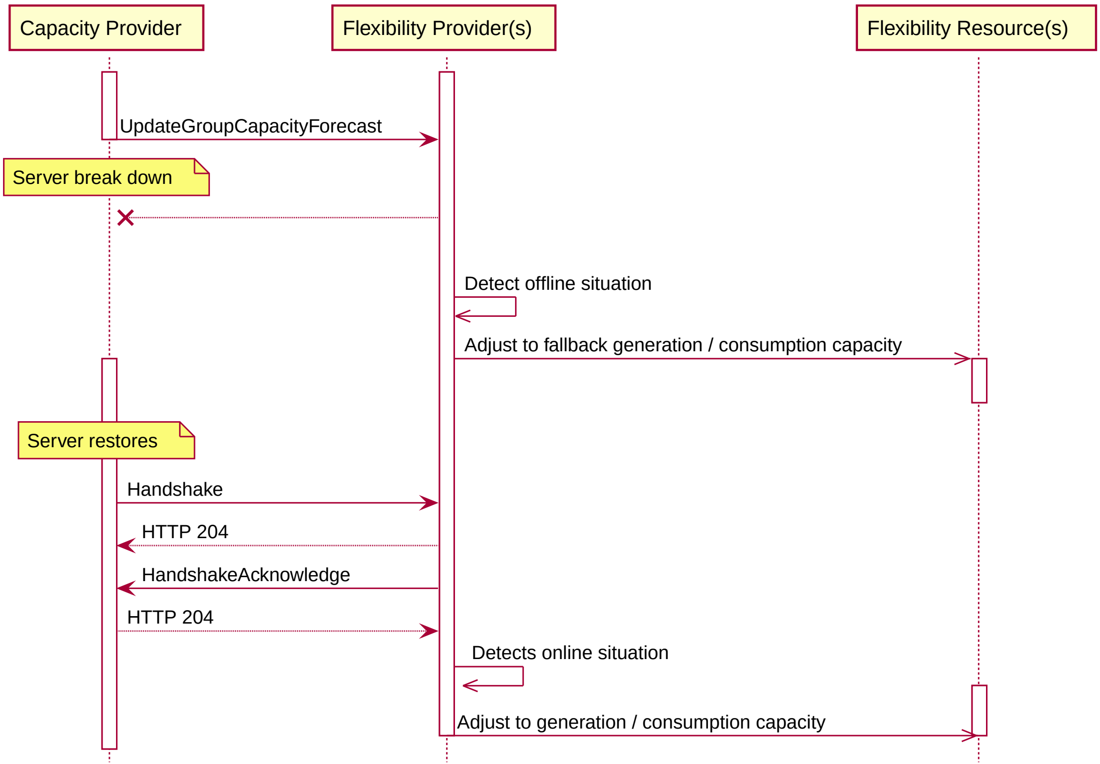


Figure 12. Sequence Diagram: Flexibility Provider adapts to offline situation

7	**Error Handling**	n/a
8	**Remarks**	n/a
9	**Example**	A charging station operator (CSO) serves as a FP and manages EV charging on a parking lot with a demand control algorithm (Smart Charging). The available Capacity is shared with DC bus chargers. The bus chargers are not controlled, meaning that they charge at full rates whenever possible. Since the total Capacity available for charging EVs and buses is limited, the CSO has to limit available Capacity for EVs when it knows the bus chargers are active. By reducing EV charging Capacity with (part of) the amount drawn by the buses, the EVs may still charge, albeit slower so that local grid constraints on the site are never exceeded. The Energy Management System (EMS) serves a a CP in the building and measures the bus chargers and informs the CSO when the Capacity for EV charging should go down. This is done by adjusting the Capacity. However, if the EMS fails to send the UpdateGroupCapacityForecast message, the CSO no longer knows whether the buses are charging. To guarantee meeting the grid constrains, the CSO has to assume the worst case: that the buses are always charging at full rates. It needs to update the EV charging Capacity accordingly. The EMS can let the CSO know in advance which Fallback Capacity to use when the communication line between EMS and CSO fails. This is the Fallback Capacity that can be sent in the Capacity forecast message. When the communication fails, the CSO automatically reduces EV charging Capacity to the Fallback Capacity that was communicated by the EMS. Note that this does not solve the case where the EMS can communicate with the CSO but not with the bus chargers. In that case the EMS should simply lower the maximum Capacity to account for the same worst case scenario, so that the CSO will not have to make any adjustments to its logic.

4. Messages

In this chapter the messages are discussed in more detail. The messages which are necessary to implement the process described in the use cases are describes here. To separate the logical protocol from the technical implementation, the protocol binding is discussed in a separate chapter.

4.1. General

The protocol is based on HTTP combined with JSON formatting (mimetype application/json). It fits within a RESTful architecture.

4.1.1. Security

The endpoints (HTTP) are protected with SSL and token based authentication. Please note that this mechanism does not require client side certificates for authentication, only server side certificates in order to set up a secure SSL connection.

4.1.2. HTTP Requests

OSCP requires a certain number of HTTP headers to be present in the request. These headers are used for authorization, message correlation and optional segmentation of a large message in several segments.

OSCP makes use of the following HTTP headers:

HTTP Header	Purpose	Mandatory
Authorization	Header with the authorization token. Example: "Authorization: Token kdHRPXGpvtPXl2KKzrKpU8zygVcfOtHGpc35VLQ4qF4"	Yes
X-Request-ID	Unique request ID of this message.	Yes
X-Correlation-ID	Reference to a request ID that this message is a response to.	Yes, for responses
X-Segment-Index	Current segment number in a segmented message. First segment has index 1.	Only for segmented messages
X-Segment-Count	Total number of segments for the complete message.	Only for segmented messages

As can be seen from the table above, every message SHALL have at least an **Authorization** header and **X-Request-ID** header.

Messages that are a response to a request, SHALL refer to that request in their **X-Correlation-ID** header.

4.1.2.1. Segmented Messages

If a message is too large to be sent in one HTTP request, then it can be sent in smaller segments. The receiver then combines all segments to get the full message.

If a message is segmented, then the headers **X-Segment-Index** and **X-Segment-Count** SHALL be present on every segment. The segment index umber is provided in **X-Segment-Index** and the total number of segments for the entire message is in **X-Segment-Count**. The segments SHALL be combined in order of **X-Segment-Index**.

For the first segment:

The first segment has **X-Segment-Index** = 1.
If the message is related to an earlier request, then it will refer to that in the header **X-Correlation-ID**.

Next segments:

All other segments SHALL refer to the **first** segment in the header **X-Correlation-ID**.

4.1.2.1.1. Example Segmented Message

Assume this message is a response to a request with X-Request-ID = 1000 and the response message is sent as 2 segments.

First segment:

```
Authorization: Token kdHRPXGpvtPXl2KKzrKpU8zygVcfOtHGpc35VLQ4qF4
X-Request-ID: 1001
X-Correlation-ID: 1000
X-Segment-Index: 1
X-Segment-Count: 2
<empty line>
<First part of message
...
...>
```

Second segment:

```
Authorization: Token kdHRPXGpvtPXl2KKzrKpU8zygVcfOtHGpc35VLQ4qF4
X-Request-ID: 1002
X-Correlation-ID: 1001
X-Segment-Index: 2
X-Segment-Count: 2
<empty line>
<Second and last part of message
...
...>
```

4.1.3. HTTP Responses

4.1.3.1. Valid Response

If an HTTP request is considered valid and no error has occurred, an HTTP response without a payload MUST be returned with HTTP status code **204**.

4.1.3.2. Error Response

In case of an error, an error response MUST be returned with a corresponding HTTP status code. The body of the error message MAY contain a human readable message explaining the cause of this message. If the error is related to an earlier message, then this message MAY be referred to in the header **X-Correlation-ID**.

During registration an error **501 Not Implemented** will be returned when none of the OSCP versions that are offered in the registration message are supported by the server.

Providing a human readable message in error situations is not mandatory, but it is RECOMMENDED to do so. Proper debugging information will help troubleshooting in development and deploy situations.

Definition Optional Error Body

Field Name	Field Type	Card.	Description
message	string	0..1	A human readable message explaining the cause of this message.

4.2. Endpoints

Every message has an endpoint that is defined relative to the base URL that was provided in the registration message. The base URL is free to be chosen by each party, but the extensions of the base URL for each message are defined in the protocol, as shown below:

OSCP Message	URL extension to base URL	Owner(s)
Register	<base>/register	CP, FP, CO
Handshake	<base>/handshake	FP, CO
HandshakeAcknowledge	<base>/handshake_acknowledge	CP, FP
Heartbeat	<base>/heartbeat	CP, FP, CO

UpdateGroupCapacityForecast	<base>/update_group_capacity_forecast	FP, CO
AdjustGroupCapacityForecast	<base>/adjust_group_capacity_forecast	CP
GroupCapacityComplianceError	<base>/group_capacity_compliance_error	CP
UpdateGroupMeasurements	<base>/update_group_measurements	CP
UpdateAssetMeasurements	<base>/update_asset_measurements	CO

IMPORTANT The above defines the standard endpoints for OSCP version 2.0. Later versions may change or add endpoints, but the endpoint for the Register message must always be the same (i.e. <base>/register) in every OSCP protocol version.

4.3. Connection

4.3.1. Registration

The endpoints are protected using token based authentication. Registration is done once by calling the *'<base>/register'* endpoint and applies to all endpoints below *'<base>'* (See Endpoints).

The diagram below shows the registration sequence.

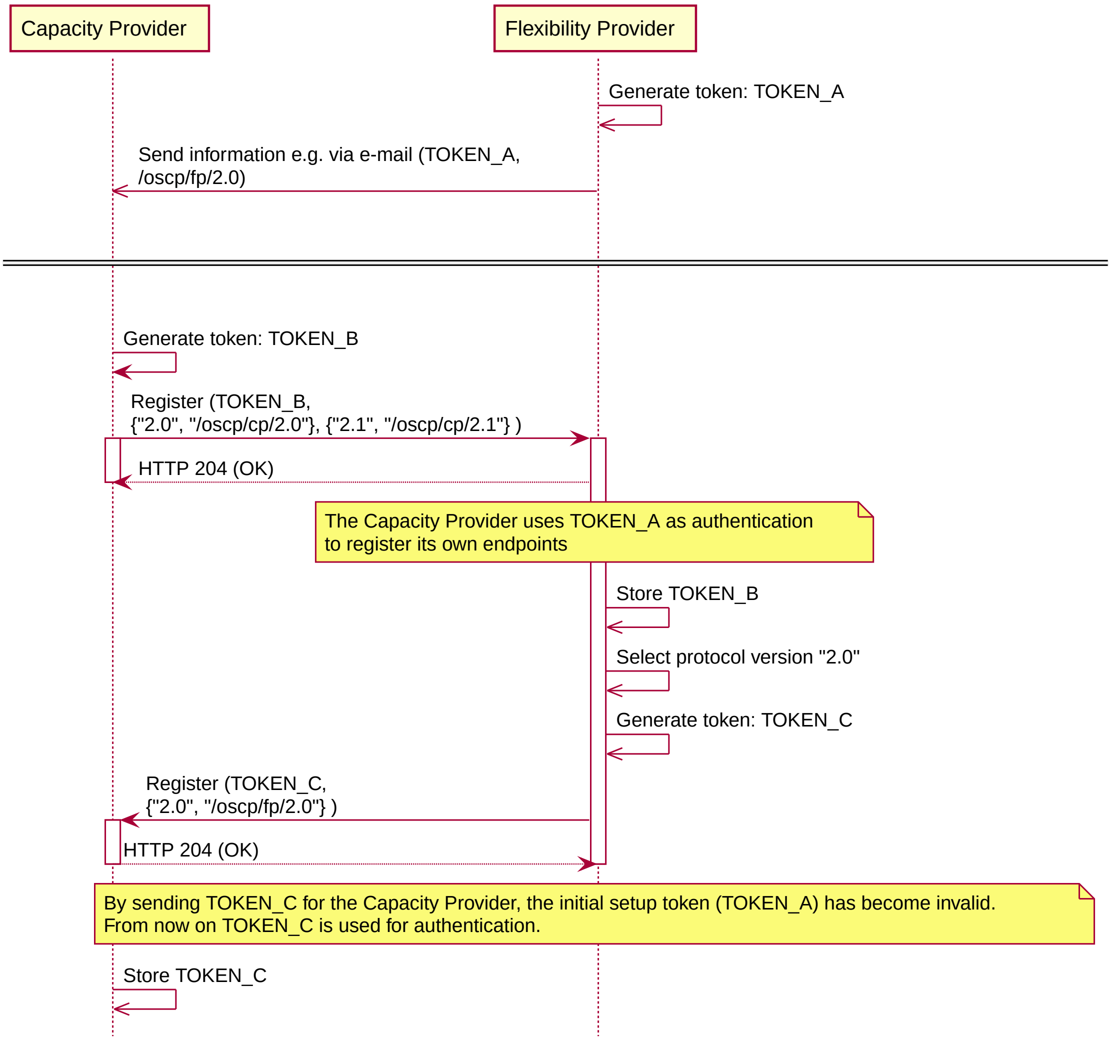


Figure 13. Sequence Diagram: Registration process

4.3.1.1. Register

The registration should be done **prior** to sending any other message. Once the registration of an endpoint is completed the security of the connection is guaranteed. The token (resulting from the registration) can be updated at any time.

Endpoints

A **POST** initiates the registration process for this base endpoint. If successful, the server must generate a new token and respond with the client's new token to access the server's system.

A **PUT** will make it possible for the client to switch communication from the server to a different client's endpoint.

A **DELETE** instantiates the un-registration process. Both parties must end any automated communication.

Endpoint	/oscp/fp/2.0/register
HTTP Method	POST, PUT, DELETE
Direction	Capacity Provider → Flexibility Provider
	Capacity Optimizer → Flexibility Provider

Endpoint	/oscp/cp/2.0/register
HTTP Method	POST, PUT, DELETE
Direction	Flexibility Provider → Capacity Provider

Endpoint	/oscp/co/2.0/register
HTTP Method	POST, PUT, DELETE
Direction	Flexibility Provider → Capacity Optimizer

Definition

Field Name	Field Type	Card.	Description
token	string	1..1	The token for the other party to authenticate in your system.
version_url	VersionURL	1..*	The initiator of the registration sends in this field the OSCP versions that it supports with associated base URLs. When used as a reply, it contains the OSCP version that is selected, with the associated base url.

4.3.2. Handshake

A handshake request is the message that initiates the handshaking mechanism. It is usually followed by a HandshakeAcknowledge. This handshake message MUST be replied to with a HTTP 204 **prior** to sending the acknowledge message.

Endpoints

Endpoint	/oscp/fp/2.0/handshake
HTTP Method	POST
Direction	Capacity Provider → Flexibility Provider

Endpoint	/oscp/co/2.0/handshake
HTTP Method	POST
Direction	Flexibility Provider → Capacity Optimizer

Definition

Field Name	Field Type	Card.	Description
required_behaviour	RequiredBehaviour	0..1	Optional. Contains configurations that define slight adjustments to the behaviour of the other party. If none is given, the other party is free to implement its own behaviour.

4.3.3. HandshakeAcknowledge

A handshake acknowledge is the message that is sent in response to an incoming Handshake message. This handshake message MUST be replied to with an HTTP 204 **prior** to sending the acknowledge message.

Endpoints

Endpoint	/oscp/cp/2.0/handshake_acknowledge
HTTP Method	POST

Direction	Flexibility Provider → Capacity Provider

Endpoint	/oscp/fp/2.0/handshake_acknowledge
HTTP Method	POST
Direction	Capacity Optimizer → Flexibility Provider

Definition

Field Name	Field Type	Card.	Description
required_behaviour	RequiredBehaviour	0..1	Optional. Contains configurations that define slight adjustments to the behaviour of the other party. If none is given, the other party is free to implement its own behaviour.

4.3.4. Heartbeat

The purpose of the heartbeat message is to periodically notify the sender's availability to the hand-shaken party. The interval in which a heartbeat is sent should be determined using the Handshaking mechanism. Therefore, sending a heartbeat is only permitted **after** the handshake is completed.

Endpoints

Endpoint	/oscp/fp/2.0/heartbeat
HTTP Method	POST
Direction	Capacity Provider → Flexibility Provider
	Capacity Optimizer → Flexibility Provider

Endpoint	/oscp/cp/2.0/heartbeat
HTTP Method	POST
Direction	Flexibility Provider → Capacity Provider

Endpoint	/oscp/co/2.0/heartbeat
HTTP Method	POST
Direction	Flexibility Provider → Capacity Optimizer

Definition

Field Name	Field Type	Card.	Description
offline_mode_at	datetime	1..1	A time in the future that indicates when, in case no more heartbeat messages are received, it can be assumed the receiving party is offline (unavailable). This time SHOULD be updated with every heartbeat message.

4.4. Capacity

4.4.1. UpdateGroupCapacityForecast

This message contains a Capacity Forecast of the capacity of a certain group (or area) for a time period. The forecast can for example be created based on measurements from a transformer or household energy consumption statistics at a certain moment in time.

The message is sent from Capacity Provider to the Flexibility Provider and from Flexibility Provider to Capacity Optimizer which should generate an Optimum capacity forecast for the capacity that should be used in the specific group.

It is based on the principle of time division, so the message contains blocks.

Endpoints

Endpoint	/oscp/fp/2.0/update_group_capacity_forecast
HTTP Method	POST

Direction	Capacity Provider → Flexibility Provider

Endpoint	/oscp/co/2.0/update_group_capacity_forecast
HTTP Method	POST
Direction	Flexibility Provider → Capacity Optimizer

Definition

Field Name	Field Type	Card.	Description
group_id	string	1..1	The id of the area in which the Flexibility Provider has Flexibility Resources connected to the grid.
type	CapacityForecastType	1..1	Identifies the type of forecast.
forecasted_blocks	ForecastedBlock	1..*	The technical content of this message.

4.4.2. AdjustGroupCapacityForecast

In case the demands of a Flexibility Provider do not match the capacity limits set by the Capacity Provider it is possible for the Flexibility Provider to request for adjustment of the capacity. If the Capacity Provider in fact decides to respond to the request it will report the updated Capacity Forecast within a UpdateGroupCapacityForecast message.

NOTE This message is for incidentally requesting adjustments to the capacity for a short time period. In case sent frequently by multiple Flexibility Providers requesting more capacity, this mechanism will lose its power.

Endpoints

Endpoint	/oscp/cp/2.0/adjust_group_capacity_forecast
HTTP Method	POST
Direction	Flexibility Provider → Capacity Provider

Definition

Field Name	Field Type	Card.	Description
group_id	string	1..1	The id of the area in which the Flexibility Provider has Flexibility Resources connected to the grid.
type	CapacityForecastType	1..1	Identifies the type of forecast.
forecasted_blocks	ForecastedBlock	1..*	The technical content of this message. Describes the amound and period of the to be adjusted capacity.

4.4.3. GroupCapacityComplianceError

This message is for notifying the Capacity Provider the Flexibility Provider cannot comply to the Capacity Forecast within a UpdateGroupCapacityForecast message.

The Capacity Forecast referred to by the Flexibility Provider SHALL be indicated by the **X-Correlation-ID** header.

Endpoints

Endpoint	/oscp/cp/2.0/group_capacity_compliance_error
HTTP Method	POST
Direction	Flexibility Provider → Capacity Provider

Definition

Field Name	Field Type	Card.	Description
message	string	1..1	A (detailed) description of the compliance error.
forecasted_blocks	ForecastedBlock	0..*	Optional. The list of forecast blocks that FP cannot comply to.

4.5. Metering

4.5.1. UpdateGroupMeasurements

This message is for communicating the total usage per aggregated area (group) from Flexibility Provider back to the Capacity Provider. This information is necessary for the Capacity Provider to know how much energy each Flexibility Provider has used according to the Capacity Forecast limits sent within the UpdateGroupCapacityForecast message.

Furthermore, the information can be used to determine a division of the Capacity Forecast over the different Flexibility Providers.

The total usage can be 'nothing'. Therefore, the *measurements* field can be empty.

Endpoints

Endpoint	/oscp/cp/2.0/update_group_measurements
HTTP Method	POST
Direction	Flexibility Provider → Capacity Provider

Definition

Field Name	Field Type	Card.	Description
group_id	string	1..1	The id of the area the Flexibility Resources (assets) are part of.
measurements	EnergyMeasurement	1..*	Contains the accumulated measurements.

4.5.2. UpdateAssetMeasurements

This message can contain various types of metering values and is send by the Flexibility Provider to the Capacity Optimizer. The Capacity Optimizer can use this information for composing an optimized profile (which in turn is sent back within an UpdateGroupCapacityForecast message).

The *measurements* field is mandatory because a message without any makes no sense. When the FP has no available measurements, for instance no running EV sessions are available, this message SHOULD NOT be sent to the Capacity Optimizer. For each Flexibility Resource (asset) multiple measurements can be made, for instance: the measuring of the assets import and export energy register or the stop of an EV session and the start of another one on that asset.

Endpoints

Endpoint	/oscp/co/2.0/update_asset_measurements
HTTP Method	POST
Direction	Flexibility Provider → Capacity Optimizer

Definition

Field Name	Field Type	Card.	Description
group_id	string	1..1	The id of the area which the Flexibility Resources (assets) are part of.
measurements	AssetMeasurement	1..*	Contains measurements of the Flexibility Resources (assets).

5. Datatypes

5.1. VersionURL

The base endpoint that is used for a specific OSCP version. Each URL MUST start with a protocol and protocol delimiter, i.e. https:// and MUST NOT end with a slash (/).

Definition

Field Name	Field Type	Card.	Description
version	string	1..1	Mandatory. The OSCP version, e.g. "2.0".
base_url	URL	1..1	Mandatory. The base url for this version, e.g. "https://oscp/cp/2.0".

5.2. RequiredBehaviour

The *behaviour of the other party* that is required for the sender to function properly.

Definition

Field Name	Field Type	Card.	Description
heartbeat_interval	integer	0..1	Optional. The interval (in seconds) in which the sender of this response expects heartbeats to receive. If provided, value must be 1 or higher. If the sender is not interested in the heartbeat of the receiver, this field can be omitted.
measurement_configuration	MeasurementConfiguration	0..*	For determining how measurements are aggregated. Providing multiple configurations is allowed. An empty array represents no configurations.

At least *one* of the above fields must be given; empty required-behaviour objects are *not* allowed.

5.2.1. MeasurementConfiguration

A unit for *measurement_configuration* of RequiredBehaviour.

Definition *Enumeration*

CONTINUOUS	The accumulated measurements are never reset. It is assumed that the asset in question keeps on accumulating its meter value from the moment it was installed, similarly to a home/office situation.
INTERMITTENT	The accumulation of the measurements are reset at certain points in time (i.e. when a charging session starts).

5.3. CapacityForecastType

Identifies types of capacity forecasts.

Definition *Enumeration*

CONSUMPTION	The maximum capacity that can be imported by the flexible source.
GENERATION	The maximum capacity that can be exported by the flexible source.
FALLBACK_CONSUMPTION	The maximum capacity that can be imported by the flexible source when Offline.
FALLBACK_GENERATION	The maximum capacity that can be exported by the flexible source when Offline.
OPTIMUM	The optimum capacity that should either be imported or exported by the flexible source.

5.4. ForecastedBlock

Definition

Field Name	Field Type	Card.	Description
capacity	decimal	1..1	The value of the forecast.
phase	PhaseIndicator	1..1	Identifies the phase that the forecast is meant for.
unit	ForecastedBlockUnit	1..1	Unit of the capacity value.
start_time	datetime	1..1	The beginning of this block.
end_time	datetime	1..1	The end of this block.

5.4.1. ForecastedBlockUnit

A unit for ForecastedBlock.

Definition *Enumeration*

A	Ampere per phase (current).
W	Watt (power).
KW	Kilowatt (power).
WH	Watt-hours (energy).
KWH	Kilowatt-hours (energy).

5.5. PhaseIndicator

An indicator of the phases on which the measurement was done. It is an indicator for various measurements and ForecastedBlock.

Definition *Enumeration*

UNKNOWN	Phase on which is measured is not known or irrelevant.
ONE	Represents measurement on phase 1.
TWO	Represents measurement on phase 2.
THREE	Represents measurement on phase 3.
ALL	Measurement represents the sum of all phases (1, 2, and 3).

5.6. AssetMeasurement

A representation of a measurement on a Flexibility Resource.

Definition

Field Name	Field Type	Card.	Description
asset_id	string	1..1	Uniquely identifies the Flexibility Resource.
asset_category	AssetCategory	1..1	Defines the type of Flexibility Resource that is measured.
energy_measurement	EnergyMeasurement	0..1	Optional. Represents a read out of an accumulative energy meter.
instantaneous_measurement	InstantaneousMeasurement	0..1	Optional. Represents an instantaneous measuring.

Either **energy_measurement** or **instantaneous_measurement** should be provided, not both.

5.6.1. AssetCategory

A category of assets that are supported. Depending on the category one or more energy flows can be measured.

Definition *Enumeration*

CHARGING	All EV supply equipment. Could be bi-directional (V2G).
CONSUMPTION	Consumption unit with all loads in the group other than charging and storage units.
GENERATION	All one-way energy generation units. Examples are PV, wind turbines, generators.
STORAGE	All stationary two-way energy storage units (batteries).

5.6.2. EnergyFlowDirection

A definition of the energy flow direction. Flows are always positive.

An energy active **import** register represents the amount of energy that has been put into the asset. The corresponding asset category in this case will be consumption. An energy active **export** shows how much energy is extracted from a generation asset. The energy active **net** is effectively the difference between import and export.

Depending on the type of asset one or more registers are available. Though net is mandatory.

Definition *Enumeration*

NET	Indicates an energy active net register.
IMPORT	Indicates an energy active import register.
EXPORT	Indicates an energy active export register.

5.6.3. EnergyMeasurement

An *energy* measurement that is by itself aggregated. An object of this type is synchronous to an accumulated energy meter reading of an aggregated area.

Definition

Field Name	Field Type	Card.	Description
value	decimal	1..1	The value of the actual measured energy.
phase	PhaseIndicator	1..1	This field identifies the phase that is measured (if applicable).
unit	EnergyMeasurementUnit	1..1	Unit of this energy value (either Wh or kWh).
energy_type	EnergyType	0..1	Indicates whether flexible, non-flexible or total energy is reported. When absent, TOTAL is assumed.
direction	EnergyFlowDirection	1..1	Indicates the direction the energy has flown into (import, export or net).
measure_time	datetime	1..1	The moment the actual meter reading took place.
initial_measure_time	datetime	0..1	Optional. The moment the measurement has (re)started (i.e. the moment an EV charge session starts). If the other party (recipient) defined a RequiredBehaviour with *INTERMITTENT* as part of the *measurement_configuration* field, then the *initial_measure_time* field MUST be set.

5.6.3.1. EnergyMeasurementUnit

A unit for EnergyMeasurement.

Definition *Enumeration*

WH	Watt-hours.
KWH	Kilowatt-hours.

5.6.3.2. EnergyType

Definition *Enumeration*

FLEXIBLE	Energy refers to the flexible load or generation.
NONFLEXIBLE	Energy refers to the non-flexible load or generation.
TOTAL	Energy refers to total load or generation.

5.6.4. InstantaneousMeasurement

A measurement that holds the current value that flows through the meter.

Definition

Field Name	**Field Type**	**Card.**	**Description**
value	decimal	1..1	The actual measured value.
phase	PhaseIndicator	1..1	This field identifies the phase that is measured.
unit	InstantaneousMeasurementUnit	1..1	Unit of the value.
measure_time	datetime	1..1	The moment the actual meter reading took place.

5.6.4.1. InstantaneousMeasurementUnit

A unit for InstantaneousMeasurement.

Definition *Enumeration*

A	Ampère per phase (current).
W	Watt (power).
KW	Kilowatt (power).
WH	Watt-hours (energy). Represents the State Of Charge in case of measuring a battery.
KWH	Kilowatt-hours (energy). Represents the State Of Charge in case of measuring a battery.