



OPEN
CHARGE
POINT
INTERFACE

OCPI 3.0-0

Introduction

<https://github.com/ocpi> & <https://evroaming.org>

document version review2, 2024-02-22

Table of Contents

1. OCPI	3
1.1. OCPI 3.0-0 Introduction	3
1.1.1. Editorial note	3
1.2. Introduction and background	3
2. Terminology and Definitions	5
2.1. Requirement Keywords	5
2.2. Abbreviations	5
2.3. Terminology	5
2.4. EV Charging Market Roles	6
2.4.1. Typical OCPI implementations per Role	7
2.4.2. Typical OCPI implementations for TCPO to ACPO communication	7
2.5. Provider and Operator abbreviation	8
2.6. Charging topology	8
3. Supported Topologies	10
3.1. Peer-to-peer	10
3.2. Multiple peer-to-peer connections	10
3.3. Peer-to-peer multiple the same roles	12
3.4. Peer-to-peer dual roles	12
3.5. Peer-to-peer mixed roles	13
3.6. Multiple peer-to-peer platforms	14
3.7. Platforms via Hub	15
3.8. Platforms via Hub and direct	16

Copyright © 2014 – 2024 EVRoaming Foundation. All rights reserved.

This document is made available under the *Creative Commons Attribution- NoDerivatives 4.0 International Public License* (<https://creativecommons.org/licenses/by-nd/4.0/legalcode>).

EVRoaming Foundation



OCPI is developed and managed by the EVRoaming Foundation. The EVRoaming Foundation is a contributor based organisation. Everyone can join the EVRoaming Foundation via <https://www.evroaming.org>

The EVRoaming Foundation strives to keep OCPI as free from IPR as possible. If you want to contribute by adding new functionality/features, you are required to send us the signed Contributor Agreement (CA) document before contributing. To get the CA, ask for it by send an e-mail to: info@evroaming.org.

Version History

Version	Date	Author	Description
3.0-review2	2024-03-12	Michel Bayings <i>EVRoaming Foundation</i>	Updated introductory texts.
3.0-review1	2024-02-22	Reinier Lamers <i>ihomer</i>	Updated to reflect current state of documentation.
3.0 Draft 0	2019-06-21	Robert de Leeuw <i>ihomer</i>	First documentation structure for OCPI 3.0, moved all existing documentation to the new documents

Document revisions There can be multiple documentation revisions of the same version of the OCPI protocol.

The newer documentation revisions of the same protocol version can never change the content of the messages: no new fields or renaming of fields. A new revision can only clarify/fix texts/descriptions and fix typos etc.

These documentation revisions (not the first) will be named: d2, d3, d4 etc.

Examples:

- OCPI 2.1.1 is a different protocol version of OCPI than OCPI 2.1.
- OCPI 2.0-d2 is the same protocol version as OCPI 2.0, but a newer documentation revision: same protocol, newer documentation.

1. OCPI

1.1. OCPI 3.0-0 Introduction

This document is the introduction to OCPI 3.0-0.

1.1.1. Editorial note

This is the first documentation structure for the new OCPI 3.0-0. This is OCPI 3.0-0-review2.

1.2. Introduction and background

The Open Charge Point Interface (OCPI) enables a scalable, automated EV roaming setup between Charge Point Operators and e-Mobility Service Providers. It supports authorization, charge point information exchange (including live status updates and transaction events), charge detail record exchange, remote commands to Charging Stations and, the exchange of smart-charging related information between parties.

It offers market participants in EV an attractive and scalable solution for (international) roaming between networks, avoiding the costs and innovation-limiting complexities involved with today's non-automated solutions or with central roaming hubs. As such it helps to enable EV drivers to charge everywhere in a fully-informed way, helps the market to develop quickly and helps market players to execute their business models in the best way.

What does it offer (main functionality):

- A good roaming system (for bilateral usage and/or via a hub).
- Real-time information about location, availability and price.
- A uniform way of exchanging data (Charging Session objects and Charge Data Records (CDRs)), before during and after the transaction.
- Remote mobile support to access any charge station without pre-registration.

Starting in 2010, e-laad foundation and the predecessor of the eViolin association specified 2 standards in order to retrieve Charging Station details and active state. These are called the VAS interface and the Amsterdam interface. In this same period, a CDR format for the exchange of charge sessions between eViolin members was defined. This format is currently in use by the majority of the eViolin members. (eViolin is the industry organisation for EV operators and service providers in Europe and responsible for national roaming and issuing of IDs). This resulted in 2014 in the development of OCPI.

OCPI is supported by a large group of organizations all over the world and is the de-facto standard for roaming and data sharing between operators, service providers, navigation providers, access points and governments. The OCPI protocol is now managed by the EVRoaming Foundation which holds the Intellectual Property Rights. Companies and organizations from all over the world can join the Foundation as a Contributor and participate in the continued development and improvement of the OCPI protocol. More information can be found at <https://www.evroaming.org/>.

OCPI is developed with support of:

evRoaming4EU project and its partners:

<https://www.evroaming4.eu>

The latest version of this specification can be found here: <https://github.com/ocpi>

2. Terminology and Definitions

2.1. 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 <https://www.ietf.org/rfc/rfc2119.txt>.

2.2. Abbreviations

Abbr.	Description
ACPO	Administrative Charging Point Operator
CDR	Charge Detail Record
CPO	Charging Point Operator
CSP	Customer Service Provider
eMSP	e-Mobility Service Provider
EV	Electric Vehicle
EVSE	Electric Vehicle Supply Equipment. This term is used in OCPI as an independently operated and managed part of a Charging Station that can deliver energy to one EV at a time.
JSON	JavaScript Simple Object Notation
LO	Location Owner
NAP	National Access Point
NSP	Navigation Service Provider
OCPP	Open Charge Point Protocol
SCSP	Smart Charging Service Provider
TCPO	Technical Charging Point Operator

2.3. Terminology

Term	Description
Charging Station	The physical system where an EV can be charged. A Charge Point has one or more EVSEs.
Platform	A software system that provides services via OCPI. A platform can provide service for a single eMSP or CPO, or for multiple CPOs or eMSPs. It can even provide services for both eMSPs and CPOs at the same time.
Pull	A system calls GET request to retrieve information from the system that owns the data.

Term	Description
Push	The system, owning the data, actively calls POST/PUT/PATCH to update other systems with new/update information.

2.4. EV Charging Market Roles

In the EV Charging landscape, different market roles can be identified.

Role	Description
ACPO	Administrative CPO: Has roaming contracts with eMSPs. Does not manage a network of Charge Points, they are using a TCPO for that. From the point of view of MSPs and Hubs, this is the CPO. The ACPO role is defined in OCPI for the use of OCPI between ACPO and TCPO only.
CPO	Charging Point Operator: Operates a network of Charge Points.
CSP	Customer Service Provider: Provides Customer Service services to parties like CPOs and eMSPs
eMSP	e-Mobility Service Provider: gives EV drivers access to charging services.
Hub	Can connect one or more CPOs to one or more eMSPs.
LO	Location Owner, owner of the location where the Charge Point is located. Typically interested in Location/Tariff information for their own locations, and the Session that happen at their locations, most of the times for making dashboards or gathering statistics.
NAP	Provides a national Database with all (public) Charging Locations, information can be send and retrieved from the NAP (that makes it different from a typical NSP).
NSP	Provides EV drivers with location information of Charge Points.
Roaming Hub	See: Hub
SCSP	Provided Smart Charging service to other parties, might use a lot of different inputs to calculate Smart Charging Profiles.
TCPO	Technical CPO. Manages a network of Charge Points for a ACPO. Does not have roaming contracts with eMSPs. From the point of view of MSPs and Hubs, the TCPO does not exist. The TCPO role is defined in OCPI for the use of OCPI between TCPO and ACPO only.

Some of these roles can be combined in one company. A Platform can provides service for multiple CPOs or eMSP, but also for both eMSPs and CPOs.

OCPI 2.0/2.1.1 had a very strict definition or roles: only CPO and eMSP. But this is rare in the real world, there are almost no parties that are strict CPO or eMSP and have there own platform. In the real-world, lots of parties provide service to CPOs that are not running their own platform. A lot of CPOs are also eMSP. With OCPI 2.1.1 and earlier that mend having to setup a OCPI connection per role.

OCPI 2.2 introduced more roles and abstracts the role from the OCPI connection. It also introduced the notion of Platforms as separate from Parties. One Platform could host multiple Parties.

OCPI 3.0 improves upon the model of OCPI 2.2 and OCPI 2.2.1 by generalizing over the different kinds of Platforms that serve multiple Parties. OCPI 3.0 has very few specification requirements that apply specifically to Hubs, instead trying to have one specification that can be used by different kinds of Platforms, whether they be operated by self-hosting Parties, SaaS providers, TCPOs serving multiple ACPOs, or Roaming Hubs.

2.4.1. Typical OCPI implementations per Role

The following table shows the typical modules implemented by the different roles. These are not required.

The table shows the typical communication role: Receiver, Sender or Both.

Modules	CPO	CSP	eMSP	Hub	NSP	NAP	SCSP	CDRs
Sender		Receiver	Both				Charging Profiles	Receiver
		Both			Sender	Commands	Receiver	Sender
Sender	Both				Credentials	Both	Both	Both
Both	Both	Both	Both	Hub Client Info	Receiver		Receiver	Sender
Receiver	Receiver	Receiver	Locations and EVSE Status	Sender	Receiver	Receiver	Both	Receiver
Both		Sessions	Sender	Receiver	Receiver	Both		
Receiver	Tariffs and Tariff Associations	Sender		Receiver	Both	Receiver	Both	

2.4.2. Typical OCPI implementations for TCPO to ACPO communication

This section is only a guideline, as different ACPO/TCPO combinations might have different ways of working together. This section is based on some examples that have been observed in the real world.

The TCPO and ACPO roles SHALL only be used in a peer-to-peer connection to each other

The following table shows the typical modules implemented by the TCPO and ACPO role. For completeness, also the CPO role has been added to the table, showing the difference between the ACPO side and CPO side of the ACPO system.

The table shows the typical communication role: Receiver, Sender or Both.

Modules	TCPO	ACPO	CPO
CDRs			Sender
Charging Profiles	Receiver	Sender	Receiver

Modules	TCPO	ACPO	CPO
Commands	Receiver	Sender	Receiver
Credentials	Both	Both	Both
Hub Client Info			Receiver
Locations	Sender	Receiver	Sender
Sessions	Sender	Receiver	Sender
Tariffs			Sender
Tokens		Sender	Receiver

Remarks for the different modules:

Module	Remark
CDRs	CDRs are generated by the ACPO based on the information send in the Sessions.
Commands	ACPO is allowed to send StartSession and ReserveNow for any Token , not limited to Tokens it owns, as ACPO can forward commands for a lot of different eMSPs.
Tariffs	Tariff information is provided by the ACPO, TCPO has no knowledge of tariffs.
Tokens	The ACPO does not forward tokens, TCPO does a real-time authorization request for all authorizations.

2.5. Provider and Operator abbreviation

In OCPI it is advised to use Contract IDs and EVSE IDs, in international agreed format. A White paper with guidelines about the format can be downloaded via the Whitepaper section of the website of EVRoaming Foundation - <https://evroaming.org/white-papers/>. The provider and the operator names, consisting of a two-character country code and a three-character company identifier, are important here. They serve to target the right provider or operator, and they need to be known up front, at least between the cooperating parties.

In several standards, an issuing authority is mentioned that will keep a central registry of known Providers and Operators. These are so-called ID Registration Offices or IDROs. The EU has legislated that CPOs and eMSPs must request and register an ID with an IDRO. Countries outside Europe do not typically have IDROs, although it is recommended to set up such an office or get in touch with the local government for support. In these cases an ID is often chosen that is not yet used by others, without central registration and coordination.

In Europe the IDROs work together through the ID Registration Repository (IDRR). More information about IDROs and IDRR can be found at <https://benelux-idro.eu/en/more-about/id-registration-repository-idrr>. This page also contains an overview of all national IDROs with links to overviews of IDs issued by them.

2.6. Charging topology

The charging topology, as relevant to the eMSP, consists of four entities:

- *Connector* is a specific socket or cable available for the EV to make use of.
- *EVSE* is the part that controls the power supply to a single EV in a single session. An EVSE may provide multiple

connectors but only one of these can be active at the same time.

- *Charging Station* is the part that is in a physically distinct enclosure with its own user interface, like a screen and/or an RFID reader and/or a set of buttons. A Charging Station provides one or more EVSEs.
- *Location* is a facility with one or more Charging Stations, operated by the same operator, located in geographical proximity at a single address.

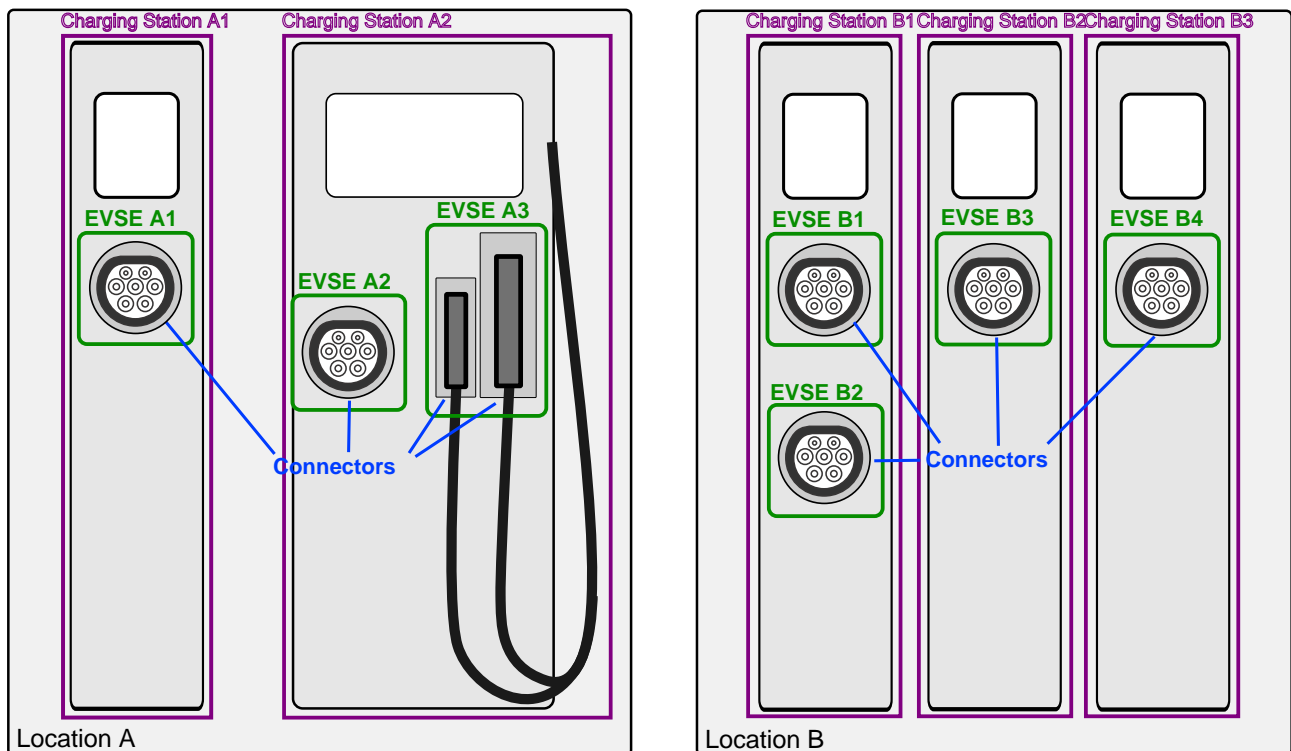


Figure 1. Charging Topology schematic

One can recognize the use of the different concepts in building EV charging solutions when you formulate an EV driver's charging user story like this:

As an EV Driver, I want to find a Location *nearby* with an EVSE that is *available* and has a Connector that is *compatible* with my car on a Charging Station that *can read my charge card*, in order to top up my car's battery to extend its range.

3. Supported Topologies

OCPI started out as a bilateral protocol, for peer-to-peer communication. Soon parties started to use OCPI via Hubs, but OCPI 2.1.1 and earlier were not designed for that. OCPI 2.2 introduces a solution for this in the form of what it called "message routing".

OCPI 2.2 introduced Platforms that connect Parties playing different roles via OCPI, instead of a fixed topology of one CPO connecting to one eMSP. More on this is written in in: [EV Charging Market Roles](#)

OCPI 3.0 preserves the separation of the Platform and Party concepts from OCPI 2.2.1, but reconsiders the "message routing" approach to Roaming Hubs. Instead OCPI 3.0 takes an approach more inspired by the proxying of HTTP servers, where the OCPI interface abstracts away the routing that might be done by a Platform in order to fulfill a request. OCPI 3.0 only gives requirements that apply to a single OCPI connection between two Platforms, that may each serve multiple Parties. This model adapts itself both to simple topologies like a connection between two self-hosting Parties and to complex topologies like Roaming Hubs or SaaS providers that serve multiple CPOs and eMSPs.

What follows is a description of all the different topologies that are supported by OCPI and were designed for in the OCPI 3.0 development process.

3.1. Peer-to-peer

The simplest topology is a bilateral connection: peer-to-peer between two Platforms, and in the most simple version each Platform hosts only one Party with only one role.

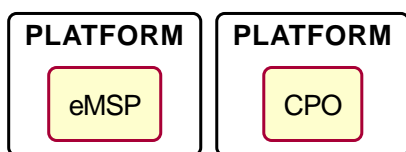


Figure 2. peer-to-peer topology example

3.2. Multiple peer-to-peer connections

A more real world topology where multiple Parties connect their Platforms, and each Party has only one role. (not every party connects with all the other parties with the other role).

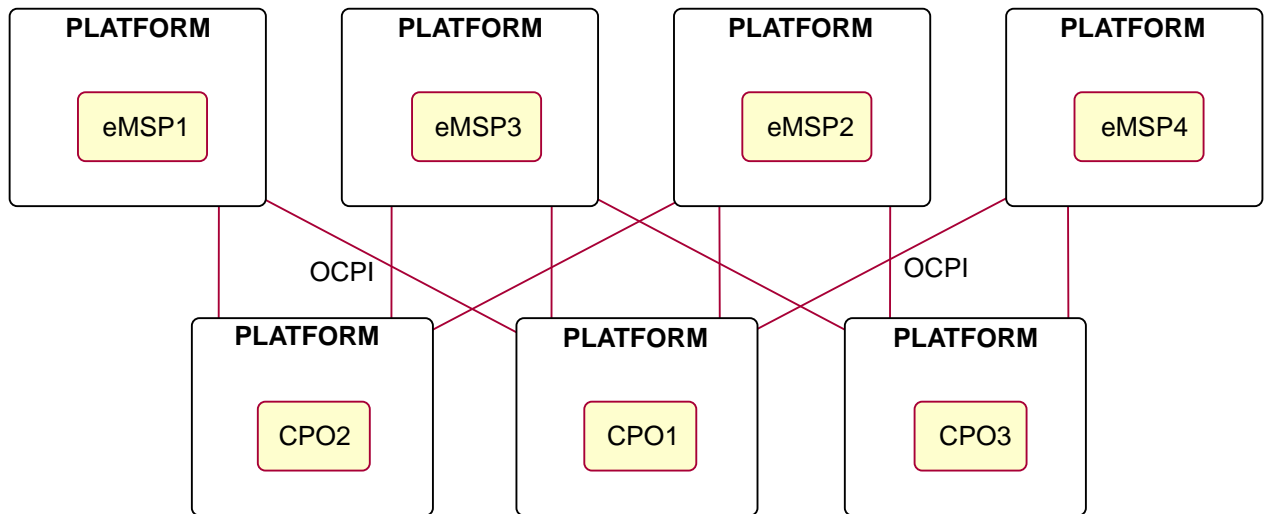


Figure 3. Multiple peer-to-peer topology example

3.3. Peer-to-peer multiple the same roles

There are companies that provide for example CPO or eMSP services for other companies. In this case, the provider company will have Platform hosts multiple Parties with the same role. This topology is a bilateral connection: peer-to-peer between two Platforms, and both Platforms host multiple Parties which have the same roles (all eMSP or all CPO).

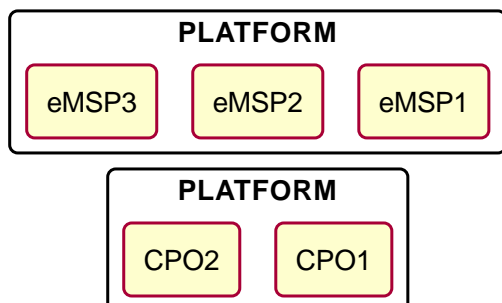


Figure 4. peer-to-peer with multiple roles topology example

3.4. Peer-to-peer dual roles

There are Parties that have dual roles, most of the companies are CPO and eMSP. This topology is a bilateral connection: peer-to-peer between two Platforms, and both Platforms have Parties with both the CPO and the eMSP roles.

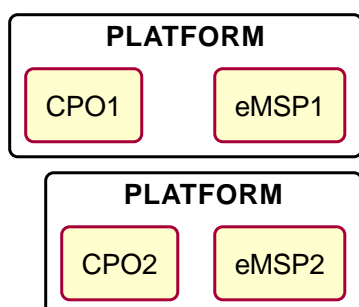


Figure 5. peer-to-peer with both CPO and eMSP roles topology example

3.5. Peer-to-peer mixed roles

There are Parties that have dual roles, or provide them to other Parties and then connect to other companies that do the same. This topology is a bilateral connection: peer-to-peer between two Platforms, and both Platforms host Parties with overlapping but different sets of roles.

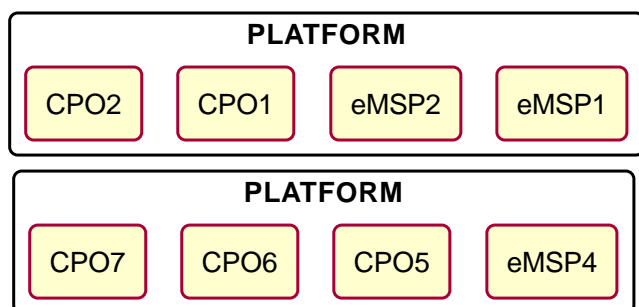


Figure 6. peer-to-peer with mixed roles topology example

3.6. Multiple peer-to-peer platforms

More a real-world topology when OCPI is used between market Parties without a hub, all Parties are hosted on their own Platforms with multiple roles.

It is easy to notice that the amount of connections between Platforms explodes.

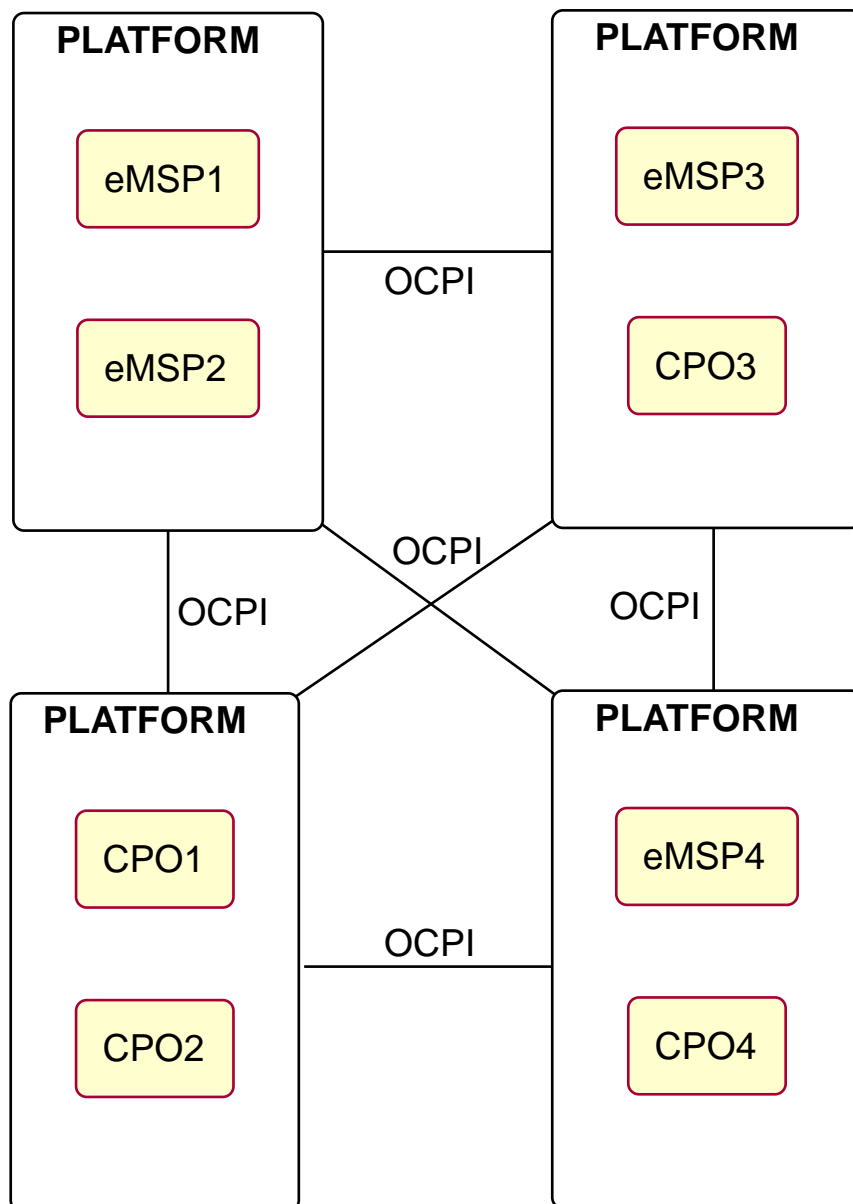


Figure 7. peer-to-peer with mixed roles topology example

3.7. Platforms via Hub

This topology has all Platforms only connect via a Hub's Platform as an intermediary. All communication goes via the Hub.

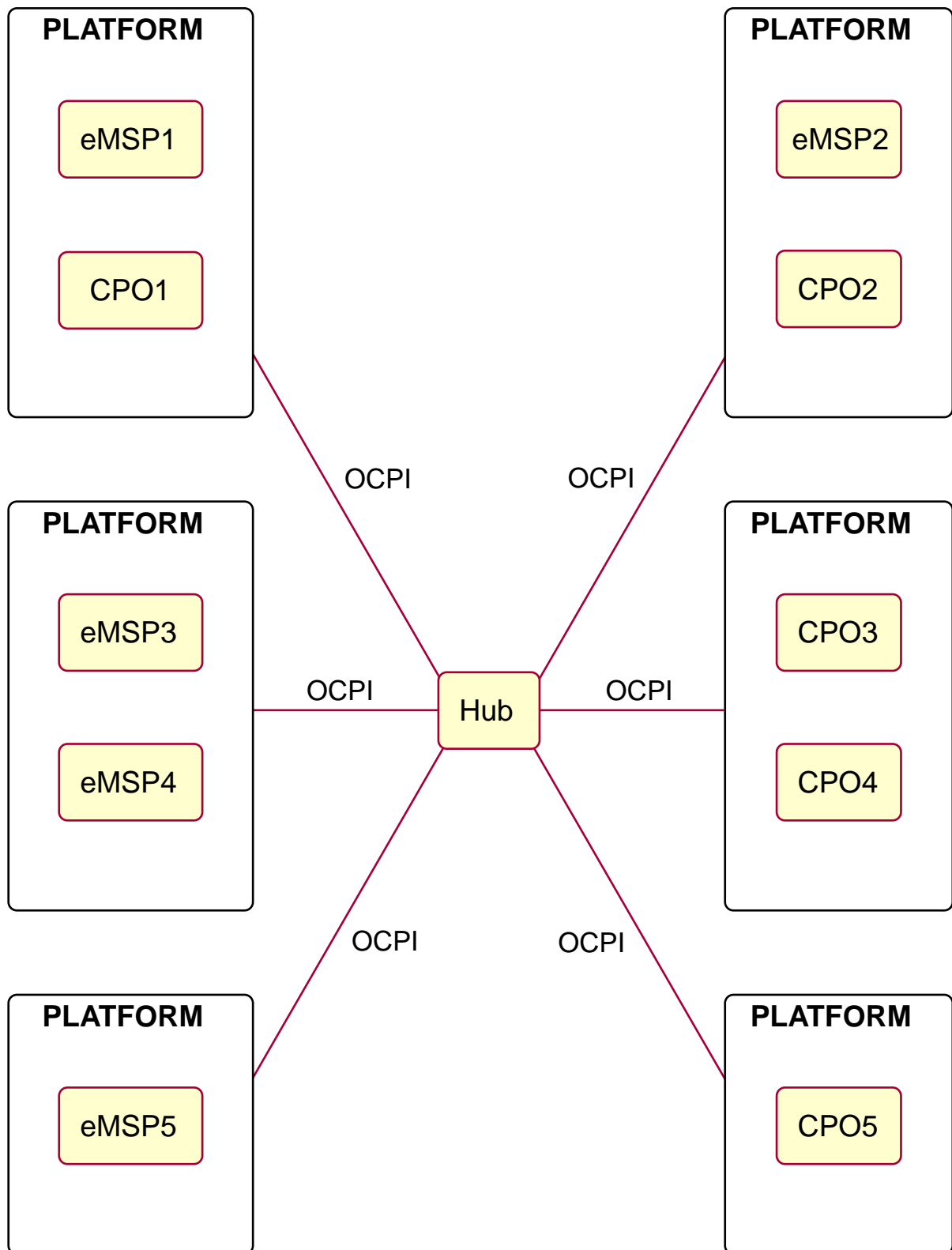


Figure 8. Platforms connected via a Hub topology example

3.8. Platforms via Hub and direct

Not all Platforms will only communicate via a Hub. There might be different reasons for Platforms to still have peer-to-peer connections.

For example:

- The Hub might not yet support new functionality.
- The Platforms use a custom module for some new project, which is not supported by the Hub.
- The Parties might not want to pay fees to use the Hub's services.
- Et cetera.

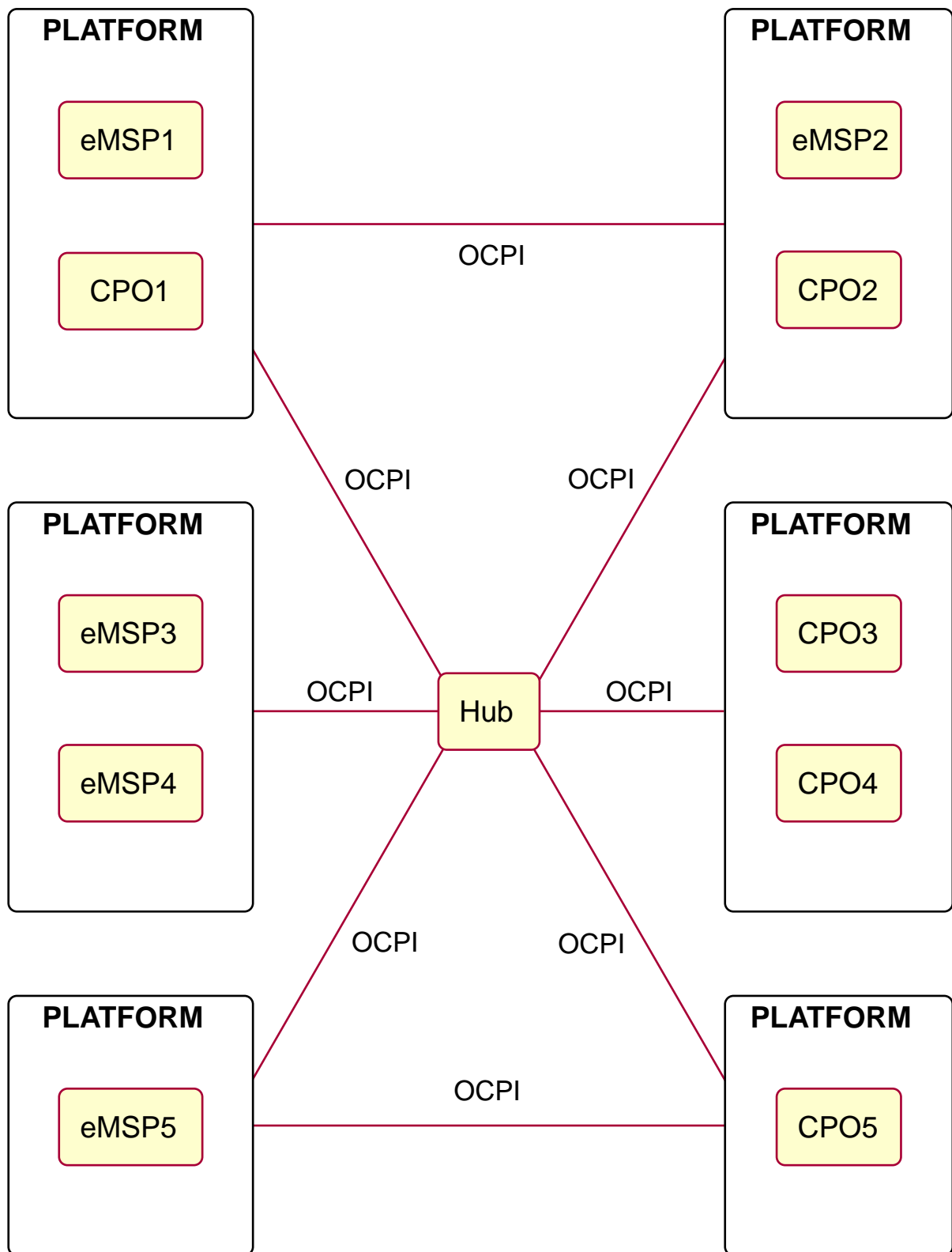


Figure 9. Platforms connected via a Hub and directly topology example