V 2.2.1

YoCharge
POWERING EV INFRASTRUCTURE

OCPI 2.2.1
Implementation Guide

The Complete Guide to Open Charge Point Interface Integration for EV Ecosystems

Interoperability | Roaming | Secure | Scalable | Future-Ready

Standardized Integration
One standard. Multiple connections.

Faster Onboarding
Integrate in days, not months.

Lower TCO
Build once, connect many.

Ecosystem Growth
Expand your network and reach.

OCPI IMPLEMENTATION GUIDE

Table of Contents

About This Guide

Written for CPO technical leads, eMSP integration engineers, EV OEM product teams, and fleet operators evaluating OCPI interoperability. Sections 01–05 are protocol-neutral; sections 06–09 are YoCharge-specific. Protocol specification reference: website.

9	3–7	10K+	25+
OCPI Modules	Days to Production	Stations via Hub	CPOs Connected

CHAPTER 01

Introduction to OCPI

OCPI (Open Charge Point Interface) is an open, royalty-free protocol maintained by the EVRoaming Foundation that enables automated data exchange between EV charging stakeholders. It solves the interoperability problem: without OCPI, every bilateral integration requires custom development, legal agreements, and ongoing maintenance between every pair of networks.

Why Interoperability Matters

As EV adoption scales, drivers expect seamless charging access across all networks — not just the network their app supports. A fleet operator running vehicles from three OEMs across six cities cannot afford separate integrations for each charging network. OCPI solves this at the protocol level: one integration, every network.

What OCPI Enables

- **EV Roaming** — drivers charge on any network
- **Station Discovery** — real-time location data
- **Driver Authentication** — token-based or app auth
- **Session Management** — remote start, stop, monitor
- **Billing & Settlement** — automated CDR exchange
- **Partner Interoperability** — one integration, many networks

OCPI vs OCPP

OCPP

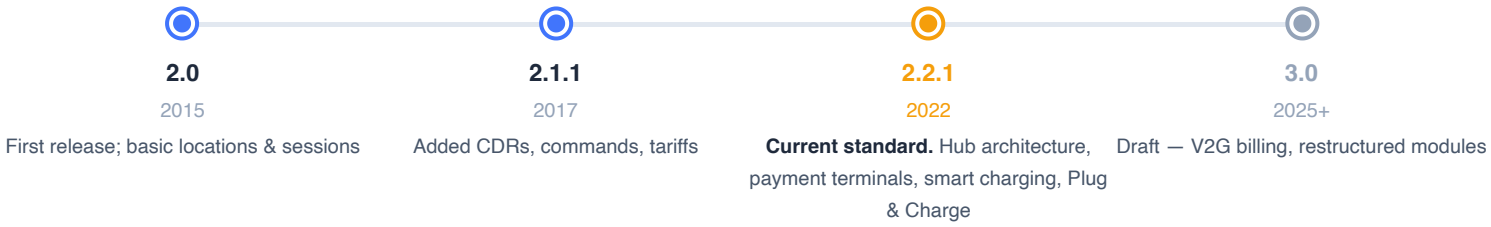
— charger ↔ backend (vertical). Governs how a physical charger talks to its management system.

OCPI

— organisation ↔ organisation (horizontal). Governs how CPOs, eMSPs, OEMs and fleets talk to each other.

Both protocols are complementary and operate independently.

Version History



OCPI 2.2.1 — What's New vs 2.1.1

HubClientInfo module (critical for hub architectures) · Payments module (direct card-on-charger) · Smart Charging profiles (grid-aware scheduling) · Enhanced token types including eMAID for Plug & Charge (ISO 15118) · Reservation flow improvements

CHAPTER 02

Key Stakeholders & Protocol Architecture

OCPI defines five primary stakeholder types. Understanding each role — and who owns which data — is essential before configuring any module.

⚡

Charge Point Operator (CPO)

Owns and operates charging stations. Shares location, EVSE status, sessions, tariffs, and CDRs. Receives token authorisation and remote commands.

📱

eMobility Service Provider (eMSP)

Issues RFID cards and apps to EV drivers. Pushes authorisation tokens to CPOs. Receives session data and CDRs for driver billing.

🚗

EV OEM

Embeds charging capability in vehicle software. Acts as an eMSP — accesses location data, remote commands, and Plug & Charge via ISO 15118 linkage.

🚛

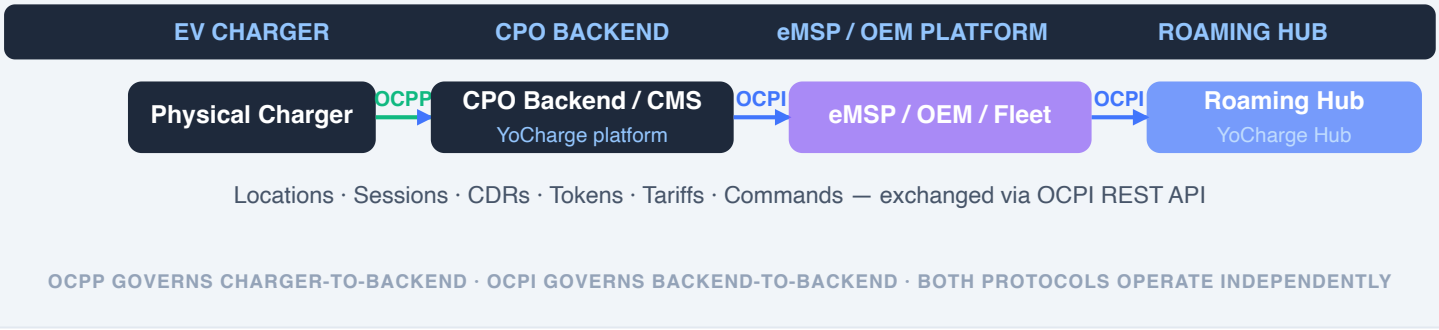
Fleet Operator

Manages EV fleets across multiple drivers and charging networks. Uses OCPI for consolidated session reporting, cost management, and multi-network billing.

Payment Terminal Provider (PTP)

— A fifth stakeholder type introduced in OCPI 2.2.1. PTPs own the physical card payment terminals installed at charging stations. They own Terminal objects and send Financial Advice Confirmations to CPOs after each transaction. See Chapter 04 for a full deep dive.

How the Protocol Stack Works



Every OCPI connection is a REST API over HTTPS. Parties exchange tokens and endpoint URLs via the **Credentials module**, then communicate module by module. Data flows are defined by the spec — the "owner" pushes data, the "receiver" pulls or acknowledges it.

CHAPTER 03

The 9 OCPI 2.2.1 Modules

OCPI is structured as independent modules. Each handles one data domain. CPOs and eMSPs negotiate which modules to exchange during the Credentials handshake — implement only what you need.

Credentials

Bidirectional

The handshake module. Exchanges API tokens, declares supported modules, versions, and endpoint URLs. Always the first module to initialise in any OCPI connection.

Locations

CPO → eMSP

Static and live data about charging stations: address, coordinates, connector types, EVSE status, and opening hours. Powers EV driver app maps.

Tokens

eMSP → CPO

eMSP pushes driver authorisation tokens (RFID, App, eMAID) to CPOs. Enables real-time or cached whitelist authorisation at the charger.

Sessions

CPO → eMSP

Live charging session data pushed by CPO during active sessions — energy consumed, duration, cost-so-far. Enables real-time updates in driver apps.

Tariffs

CPO → eMSP

Pricing structures shared by CPO. Supports time-of-use, per-kWh, flat fees, idle charges, and reservation fees.

CDRs

CPO → eMSP

Charge Detail Records — the final billing record for each session. Transferred post-session for invoicing and reconciliation.

Commands

eMSP → CPO

Remote session control: StartSession, StopSession, UnlockConnector, ReserveNow. Enables app-initiated charging without RFID.

HubClientInfo

Hub → Parties

New in 2.2.1. Hub registry of connected parties and their status. Essential for hub architectures — parties discover each other via the hub.

Payments

PTP → CPO

New in 2.2.1. Physical card payment terminals at chargers. Manages Terminal objects, maps terminals to EVSEs, settles via Financial Advice Confirmation.

Module Direction Rule

Each module has a designated data owner who pushes, and a receiver who pulls or acknowledges. Direction is defined by the OCPI spec — not by implementation preference. The Credentials module is always bidirectional — both parties must complete the handshake before any other module can be used.

CHAPTER 04

Payment Module Deep Dive

The **Payments module** (OCPI 2.2.1) enables physical card terminals at EV chargers to integrate with CPO backends without custom code. It bridges the Payment Terminal Provider (PTP), the CPO, and the roaming network — making ad-hoc card-on-charger sessions interoperable across networks.

Terminal Object

The **Terminal** maps one physical payment device to a location and its EVSEs:

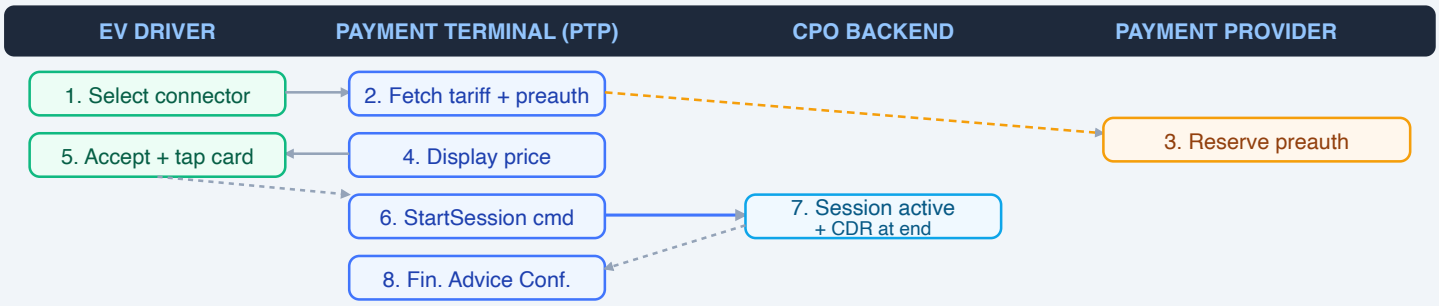
Field	Purpose
`terminal_id`	Unique device identifier
`location_ids`	Assigned charging locations
`evse_uids`	Specific EVSEs served
`invoice_base_url`	Invoice download URL
`invoice_creator`	Who generates invoices

Financial Advice Confirmation

Sent by PTP to CPO after payment capture, linking the financial record to the charging session:

Field	Purpose
`authorization_reference`	Maps to StartSession
`total_costs`	Captured amount + VAT
`capture_status_code`	SUCCESS / PARTIAL / FAILED
`currency`	ISO-4217 code
`eft_data`	EFT data for invoice

Payment Terminal Session Workflow



PTP OWNS TERMINAL · CPO OWNS SESSION & CDR · FINANCIAL ADVICE CONFIRMATION RECONCILES THE PAYMENT TO THE CHARGING RECORD

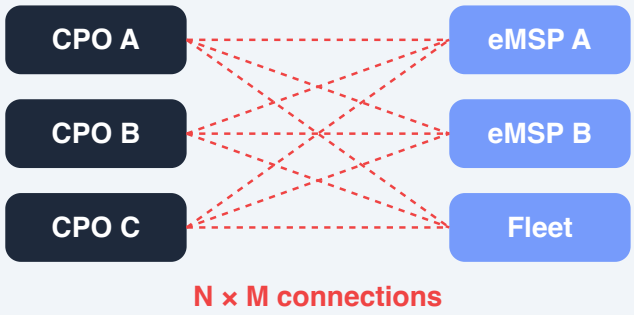
YoCharge Payment Support

YoCharge's OCPI platform supports the full Payments module. Terminal-to-EVSE mapping is managed via the YoCharge dashboard and Financial Advice Confirmations are automatically reconciled against session CDRs for seamless billing.

CHAPTER 05

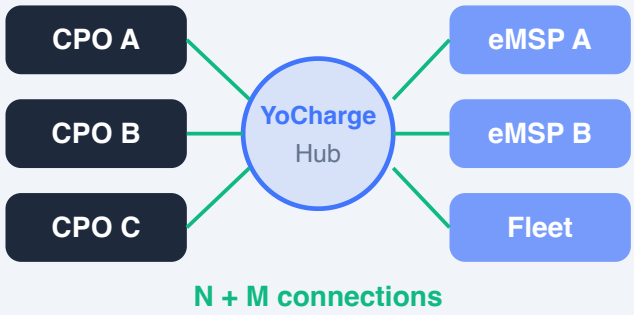
Integration Models & Session Lifecycle

Model A — Direct P2P



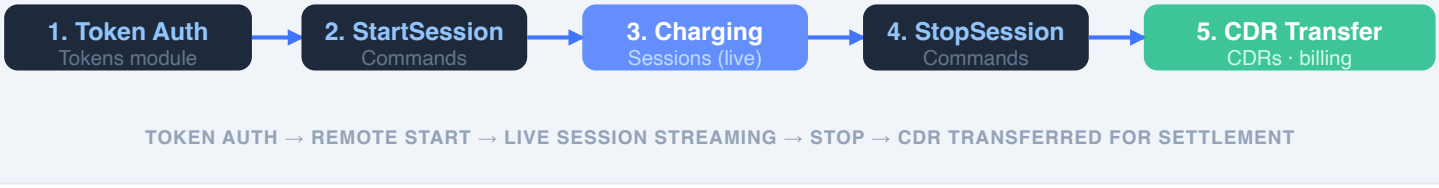
Scales poorly. 5 CPOs + 5 eMSPs = 25 integrations. Each new partner needs full custom work.

Model B — Roaming Hub (Recommended)



Scales effortlessly. Integrate once, reach all connected partners. Adding partner #10 costs zero additional dev.

Session Lifecycle — Auth to CDR



CHAPTER 06

YoCharge OCPI Platform

YoCharge delivers production-ready OCPI 2.2.1 integration capabilities across all 9 modules for CPOs, eMSPs, EV OEMs, and roaming platforms. The platform supports both direct P2P integrations and the **YoCharge Roaming Hub** — India's fastest-growing OCPI hub with 10,000+ stations across 25+ CPOs.

10K+	25+	20+	3–7
Charging Stations	CPOs Connected	Aggregators & OEMs	Days to Go Live

⚡ For CPOs

- Share stations across 20+ aggregators
- Works with or without YoCharge CMS
- Automated CDR + invoicing
- Increase network utilisation

📱 For eMSPs & OEMs

- Single integration → 10,000+ stations
- Real-time location + EVSE status
- Remote session control via app
- White-label options available

🚛 For Fleets

- Multi-network access in one platform
- Consolidated billing per vehicle
- ISO 15118 Plug & Charge ready
- Fleet cost tracking + reports

OCPI vs Custom Integration — The Business Case

Dimension	Custom API per Partner	OCPI via YoCharge Hub
First integration	4–12 weeks dev time	**3–7 days**
Adding partner #10	Full new integration	**Zero additional dev**
Maintenance overhead	Per-partner updates	**Single protocol update**
Billing reconciliation	Custom per partner	**Automated via CDRs**
Payment terminal support	Custom development	**OCPI Payments module**

Connected Partners (Sample)

CPOs: Chargezone, Statiq, Reliable Charge, Plugzsmart, ChargeMode, Mobilane, Kazam, ChargeIn + 17 more · eMSPs & OEMs: Mahindra, Audi, Maruti, Vinfast, Statiq, Ionage, Pulse Energy, Electreefi + 12 more aggregators

CHAPTER 07

Implementation Journey

YoCharge's OCPI onboarding runs in five phases. Technical setup typically completes in **3–7 business days** — from credentials exchange to live data flow.

1 Discovery & Scoping

Day 1

Define stakeholder type, required modules, and connectivity model (P2P or Hub). YoCharge team conducts requirements assessment.

2 Credentials Exchange

Day 1–2

HTTPS endpoints configured. API tokens exchanged. Both parties declare supported modules and endpoint URLs via the Credentials module.

3 Module Configuration

Day 2–4

Enable Locations, Tokens, Sessions, Tariffs, CDRs, Commands — and optionally Payments. YoCharge provides a sandbox environment.

4 Integration Testing

Day 4–6

End-to-end validation: location accuracy, token auth response times, session lifecycle, CDR accuracy, and command latency.

5 Production Deployment

Day 6–7

Cutover to live environment. YoCharge monitors the first 48 hours and provides on-call support for any data anomalies or edge cases.

What you need before starting

A publicly accessible HTTPS endpoint · API credentials from your system provider · Knowledge of required modules · A technical contact for JSON/REST API configuration

CHAPTER 08

Best Practices

These practices are drawn from YoCharge's experience running OCPI integrations across 25+ CPO networks and 20+ aggregators. Adopt them early — they are significantly harder to retrofit after go-live.

Adopt OCPI 2.2.1 as Baseline

Always implement OCPI 2.2.1 — not 2.1.1. The HubClientInfo module (2.2.1 only) is essential for hub-based roaming, and the Payments module cannot be used on older versions. Starting at 2.2.1 avoids a forced migration later.

Secure Credential Exchange

Use HTTPS with valid TLS certificates on all OCPI endpoints. Rotate API tokens on a defined schedule (typically every 90 days). Never share OCPI credentials over unencrypted channels. Store tokens in a secrets manager — not in code.

Monitor Session Synchronisation

Sessions pushed by CPOs must be received and acknowledged by eMSPs within spec-defined windows. Set up alerts for session sync lag > 60 seconds. Unacknowledged sessions are the most common source of billing disputes in live OCPI environments.

Maintain Tariff Consistency

Ensure Tariff objects match the actual pricing your chargers apply. Tariff-CDR mismatches trigger reconciliation failures and driver complaints. Push tariff updates proactively — before the effective date, not after sessions are billed at the wrong rate.

Implement Robust Observability

Log every OCPI API call — request, response, status code, and timestamp. Maintain logs for at least 90 days for dispute resolution. A CDR dispute 6 weeks after a session requires the original session and tariff data to be retrievable.

Test All Edge Cases Before Go-Live

Test: session that starts but never ends · CDR with zero kWh · command rejected by charger · tariff with multiple price components · token not found response. These edge cases account for 80% of integration issues found in production.

Common Integration Pitfalls

Pitfall	Impact	Prevention
Skipping the Credentials handshake test	Silent auth failures in production	Full handshake test before enabling other modules
Hardcoding token values in app	Security breach on token rotation	Store tokens in a secrets manager; use environment variables
Not syncing locations before session test	Session references unknown EVSE	Complete location sync before token or session testing
Ignoring CDR status codes	Billing reconciliation errors	Verify CDR `status` field; only invoice on `ACCEPTED`

READY TO IMPLEMENT OCPI?

Start your OCPI integration in 3–7 days.

Our integration team handles setup, module configuration, and go-live support. You focus on growing your network.

📅 Book a Demo — calendly.com/demo_yocharge/45min

🌐 yocharge.com/software/ocpi-integration/

Contact & Support

📞

SALES

+91-9462878912

sales@yocharge.com

WhatsApp: wa.me/919462878912

🛠️

TECHNICAL SUPPORT

+91-9079376241

support@yocharge.com

WhatsApp: wa.me/919079376241

💼

CAREERS

We're hiring

careers@yocharge.com

🔗

FOLLOW US

LinkedIn · Instagram · YouTube

linkedin.com/company/yocharge

🏢 Office Address

1st Floor, Eighty-two Spaces, Near HDFC Bank, Airport Road

Sunderwas, Udaipur, Rajasthan, India — 313001

⚡

YoCharge

EV Charging Software for CPOs

yocharge.com

🚗

YoMobility

EV Fleet Management Platform

yomobility.com

🌐

EV Roaming Hub

India's largest OCPI Roaming Hub

EV Roaming Hub