Standardized Protocol for Real-Time APIs as Required by Title 23 CFR 680.116(c)

Brennan Harris

Idaho National Laboratory

Kang-Ching (Jean) Chu, Nathan Legault, Lissa Myers, K. Shankari, and Eric Wood

Joint Office of Energy and Transportation

Version 1.0 – September 2024

It is the authors' expectation that this document will be updated as data sharing protocols between charging networks and third-party software developers mature. Readers are encouraged to consider whether it is appropriate to cite a specific version of this document or the most recent release.

Except for the statutes and regulations cited, the contents of this document do not have the force and effect of law and are not meant to bind applicants in any way. This document is intended only to provide information regarding existing requirements under the law or agency policies.

DOE/EE-2895

List of Acronyms

API	application programming interface
OCPI	Open Charge Point Interface
REST	representational state transfer

Executive Summary

Improving the ability of drivers to easily locate working and available chargers is key to improving the public charging experience. Electric vehicle charging providers who are recipients of federal funds through the National Electric Vehicle Infrastructure (NEVI) Formula Program, Charging and Fueling Infrastructure (CFI) Discretionary Grant Program, and other funding programs as identified under Title 23 of the U.S. Code must deploy and maintain an application programming interface (API) to access information about charging stations they operate.[1] This includes information about individual charging ports, pricing, and availability in accordance with the Federal Highway Administration's National Electric Vehicle Infrastructure Standards and Requirements, 23 CFR 680.116(c), herein referred to as the minimum standards (Federal Highway Administration 2023). Specifically outlined in the minimum standards, states and other designated recipients are required to ensure that charging station information including location, connector type, power level, real-time status, and real-time price to charge are available free of charge to third-party software developers through an API. These requirements are intended to enable effective communication with consumers about available charging stations and help consumers make informed decisions about trip planning, including when and where to charge.

This document provides a standardized protocol for how to structure data, data update frequency, and practices for making the data required to be shared via API usable for improving public transparency and the customer experience. These are recommendations only and do not modify the Federal Highway Administration's minimum standards in any way. A summary of these recommendations is shown below:

WHAT data are to be shared:

- Attributes required by 680.116(c) consistent with Open Charge Point Interface (OCPI) 2.2.1 formatting plus "charging station status" and "pull-through" attributes.
- File type: .json

WHEN data are to be shared:

- API is to be available simultaneous to station becoming operational.
- Data update frequency (latency):
  - Static fields: Daily.
  - Dynamic fields: Within 15 seconds.
- Average response time: 1,000 milliseconds.
- Rate limiting: 1,000 requests per hour.

[1] For a full description of the applicability of the minimum standards, refer to Section 680.102 (Federal Highway Administration 2023).

- API uptime: 99.5%.
    - 3.6 hours of downtime per calendar month.

WHO can access data:

- API documentation to be posted online by charging network.
- Any third-party software developer can request token access.
    - Networks should use reasonable discretion when granting and revoking tokens.
- Expected registered users: 10,000.
    - 1,500 simultaneous requests.

HOW data are to be shared:

- API interface: Follow representational state transfer (REST) principles (IBM 2023).
- Management: Follow API Umbrella (National Renewable Energy Laboratory 2023).
- Architecture: GET/PATCH.

Table of Contents

List of Figures

List of Tables

Expected Data Flow

Content made available by an application programming interface (API) is expected to be aggregated by third-party software developers, who will ultimately offer mapping services and other service applications to drivers (Figure 1).

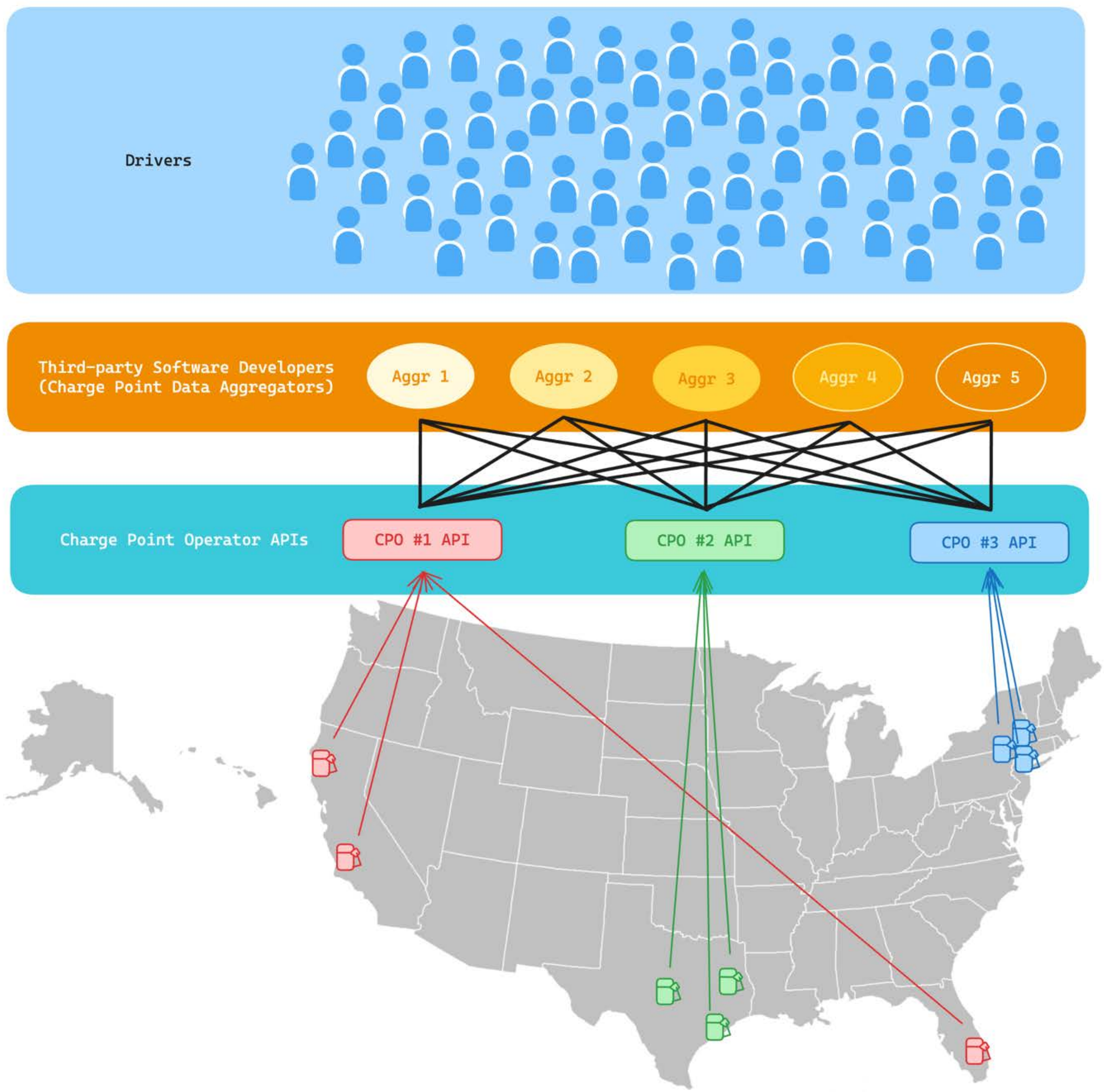

Figure 1. Expected data flow from charge point operators to aggregators and drivers

WHAT Data Are To Be Shared

Examples for most data fields can be found in the data specifications for the Alternative Fuels Data Center (2024) and the Open Charge Point Interface (OCPI) 2.2.1 (EVRoaming Foundation 2021). Table 1 summarizes the references from data in this guide to data in those two specifications.

Table 1. A Mapping of the Data Properties Referenced in This Guide to the Alternative Fuels Data Center and OCPI Data Schemas

Property	Alternative Fuels Data Center	OCPI	OCPI Reference
Station identifier	id	Location: id	8.3.1
Address	street_address	Location: address	8.3.1
	intersection_directions	Location: directions	8.3.1
	city	Location: city	8.3.1
	state	Location: state	8.3.1
	zip	Location: postal_code	8.3.1
	plus4		
	country	Location: country	8.3.1
		Location: related_locations	8.3.1
		EVSE: floor_level	8.3.2
Coordinates	geocode_status	Location: coordinates	8.3.1
	latitude		
	longitude		
Operator name	*Not present*	Location: operator: name	8.3.1
Network provider name	ev_network	Location: party_id	8.3.1
Station status	*Not present*	*Not present*	-
Access type	access_code	*Not present (OCPI assumes public)*	
Hours of operation	access_days_time	Location: opening_times	8.3.1
Number of ports	ev_level1_evse_num	Location: evses (must count)	8.3.1
	ev_level2_evse_num		
	ev_dc_fast_num		
Unique port identifier	*Not present*	EVSE: uid	8.3.2
Connector types by port	ev_connector_types	Connector: standard	8.3.3
		Connector: format	8.3.3
Charging level by port	*Not present*	Connector: power_type	8.3.3

Property	Alternative Fuels Data Center	OCPI	OCPI Reference
Power delivery by port	*Not present*	Connector: max_electric_power	8.3.3
Accessibility to trailered vehicles	*Not present*	*Not present*	-
Real-time port status	*Not present*	EVSE: status	8.3.2
Pricing structure	ev_pricing (description only)	Tariff	11.3.1
Real-time port price	*Not present*	PriceComponent	11.4.2
Payment methods accepted	cards_accepted	Capability: CHIP_CARD_SUPPORT	8.4.3
		Capability: CONTACTLESS_CARD_SUPPORT	8.4.3
		Capability: CREDIT_CARD_PAYABLE	8.4.3
		Capability: DEBIT_CARD_PAYABLE	8.4.3
		Capability: PED_TERMINAL	8.4.3
		Capability: RFID_READER	8.4.3

Definition of Terms

Charging station, as defined in 23 CFR 680.104, means the area in the immediate vicinity of a group of chargers and includes the chargers, supporting equipment, parking areas adjacent to the chargers, and lanes for vehicle ingress and egress. A charging station could comprise only part of the property on which it is located.

Charging port, as defined in 23 CFR 680.104, means the system within a charger that charges one electric vehicle at a time. A charging port may have multiple connectors, but it can provide power to charge only one vehicle through one connector at a time.

Data Specified

All of the following data are recommended for conformance with 23 CFR 680. Any example data included beyond the following list are for convenience and clarity, and not specified by this guide.

Station Data

Unique charging station name or identifier. This is a unique identifier that indicates a single charging station. It must be unique in the context of its paired network ID (party_id), and it must match the station ID utilized for 23 CFR 680.112 data submission requirements.

Address. Where possible, the address should contain street number and name, city, state, and ZIP code that specify the location of the charging station.

Geographic coordinates. Indicates the latitude and longitude in decimal degree of the exact charging station location, to 6-decimal precision. This level of precision equates to about 10 centimeters of accuracy.

Charging station operator name. The name of the entity that owns the chargers and supporting equipment and facilities at one or more charging stations.

Charging network provider name. The name of the entity that operates the digital communication network that remotely manages the chargers.

Charging station status. Indicates whether the station is currently accessible to electric vehicles to charge. The available status values are *OPERATIONAL*, *UNDER CONSTRUCTION*, *PLANNED*, and *DECOMMISSIONED*.

Charging station access type. Indicates whether charging at the station is accessible to the public or limited to specified users such as individual fleets or commercial vehicles. Valid values are PUBLIC and PRIVATE.

Charging station access days and times. Indicates the days and hours of access to charging at the station. This should be sufficiently descriptive for planning charging around station closures due to daily schedule, planned downtime periods, and exceptional closures and openings, as applicable. Valid values are displayed in OCPI 2.2.1 Section 8.3.1.

Port Data

Number of ports. The total number of charging ports available at the station.

Unique port identifier. A station-unique identifier to describe each individual charging port. The port ID must match the port ID utilized for 23 CFR 680.112 data submission requirements.

Connector types available by port. Each port should indicate which supported charging connector types are supported. The values in Table 2 are valid connector types enumerated in OCPI 2.2.1.

Table 2. Valid Values and Descriptions of Connector Types From OCPI 2.2.1

Value	Description
CHADEMO	The connector type is CHAdeMO
DC CHAOJI	The ChaoJi connector, the new generation charging connector, harmonized between CHAdeMO and GB/T, DC
DOMESTIC_A	Standard/domestic household, type "A," NEMA 1-15, 2 pins
DOMESTIC_B	Standard/domestic household, type "B," NEMA 5-15, 3 pins
DOMESTIC_C	Standard/domestic household, type "C," CEE 7/17, 2 pins
DOMESTIC_D	Standard/domestic household, type "D," 3 pins
DOMESTIC_E	Standard/domestic household, type "E," CEE 7/5, 3 pins

Value	Description
DOMESTIC_F	Standard/domestic household, type "F," CEE 7/4, Schuko, 3 pins
DOMESTIC_G	Standard/domestic household, type "G," BS 1363, Commonwealth, 3 pins
DOMESTIC_H	Standard/domestic household, type "H," SI-32, 3 pins
DOMESTIC_I	Standard/domestic household, type "I," AS 3112, 3 pins
DOMESTIC_J	Standard/domestic household, type "J," SEV 1011, 3 pins
DOMESTIC_K	Standard/domestic household, type "K," DS 60884-2-D1, 3 pins
DOMESTIC_L	Standard/domestic household, type "L," CEI 23-16-VII, 3 pins
DOMESTIC_M	Standard/domestic household, type "M," BS 546, 3 pins
DOMESTIC_N	Standard/domestic household, type "N," NBR 14136, 3 pins
DOMESTIC_O	Standard/domestic household, type "O," TIS 166-2549, 3 pins
GBT_AC	Guobiao GB/T 20234.2 AC socket/connector
GBT_DC	Guobiao GB/T 20234.3 DC connector
IEC_60309_2_single_16	IEC 60309-2 industrial connector, single phase, 16 A (usually blue)
IEC_60309_2_three_16	IEC 60309-2 industrial connector, three phases, 16 A (usually red)
IEC_60309_2_three_32	IEC 60309-2 industrial connector, three phases, 32 A (usually red)
IEC_60309_2_three_64	IEC 60309-2 industrial connector, three phases, 64 A (usually red)
IEC_62196_T1	IEC 62196 Type 1 "SAE J1772"
IEC_62196_T1_COMBO	Combo Type 1 based, DC
IEC_62196_T2	IEC 62196 Type 2 "Mennekes"
IEC_62196_T2_COMBO	Combo Type 2 based, DC
IEC_62196_T3A	IEC 62196 Type 3A
IEC_62196_T3C	IEC 62196 Type 3C "Scame"
NEMA_5_20	NEMA 5-20, 3 pins
NEMA_6_30	NEMA 6-30, 3 pins
NEMA_6_50	NEMA 6-50, 3 pins
NEMA_10_30	NEMA 10-30, 3 pins
NEMA_10_50	NEMA 10-50, 3 pins
NEMA_14_30	NEMA 14-30, 3 pins, rating of 30 A
NEMA_14_50	NEMA 14-50, 3 pins, rating of 50 A
PANTOGRAPH_BOTTOM_UP	Onboard, bottom-up pantograph typically for bus charging
PANTOGRAPH_TOP_DOWN	Offboard, top-down pantograph typically for bus charging
TESLA_R	Tesla connector "Roadster" type (round, 4 pins)
TESLA_S	Tesla connector "Model S" type (oval, 5 pins)

Charging level by port. Each port should indicate which charging level it provides and the voltage range it is capable of. The charge levels are defined as AC Level 1 (120 V), AC Level 2 (208–240 V), and DC fast (400–1,000 V) (U.S. Department of Transportation 2023).

Power delivery. The maximum power delivery in kilowatts, defined for each port.

Accessibility by vehicle with trailer. Indicates whether the port is accessible by vehicles with a trailer attached in the station design (e.g., pull-through stall). This is a Boolean value.

Real-time port status. The live status of a port indicating whether it is available for charging. This should use the display format indicated in OCPI 2.2.1.

Pricing

Pricing structure. This should indicate the way that customer charging is handled. OCPI 2.2.1 Section 11.3.1.1 contains examples of fully explicated pricing descriptions.

Real-time port price. This should indicate, at that moment in time, the price for electricity to charge ($/kWh) on each port. This should use the display format indicated in OCPI 2.2.1.

Payment methods accepted. This should indicate the full list of payment methods accepted for payment at the station.

Note that other fees necessary to calculate final charging price may be necessary for inclusion in the pricing structure that are not explicitly mentioned here.

Message Format

The API response bodies and patch updates should be returned and sent with “application/json” content type and follow the JSON data interchange syntax (Ecma International 2017). JSON is the most widely used format in representational state transfer (RESTful) services and is also simpler and more human-readable than alternative formats such as XML.

Example

An example JSON response to a hypothetical API request showing a compliant presentation of the API data for one charging station can be found below:


```
{
    "charge_points": [
        {
            "id": "decfe67d-ddc6-4bd2-bb8f-30627f10aa81",
            "address": {
                "street_address": "1266 Craters Loop Road",
                "directions": "Follow arrows directing one-way traffic flow into parking
lot.",
                "city": "Arco",
                "state": "ID",
                "postal_code": "83213",
```

```
        "country": "United States of America",
        "related_locations": [],
        "floor_level": null
    },
    "coordinates": {
        "latitude": "43.46186",
        "longitude": "-113.56206"
    },
    "operator_name": "ChargeOperator LLC",
    "provider_name": "ChargeProvider Incorporated",
    "station_status": "OPERATIONAL",
    "access_type": "PUBLIC",
    "station_schedule": {
        "twentyfourseven": false,
        "time_zone": "US/Mountain",
        "open_hours": [
            {
                "weekday": 1,
                "period_begin": "08:00",
                "period_end": "17:00"
            },
            {
                "weekday": 2,
                "period_begin": "08:00",
                "period_end": "17:00"
            },
            {
                "weekday": 3,
                "period_begin": "08:00",
                "period_end": "17:00"
            },
            {
                "weekday": 4,
                "period_begin": "08:00",
                "period_end": "17:00"
            },
            {
                "weekday": 5,
                "period_begin": "08:00",
                "period_end": "17:00"
            },
            {
                "weekday": 6,
                "period_begin": "10:00",
                "period_end": "14:00"
            }
        ],
        "exceptional_openings": [
            {
                "period_begin": "2025-09-23T08:00:00Z",
                "period_end": "2025-09-23T10:00:00Z"
            },
            {
                "period_begin": "2025-09-23T14:00:00Z",
                "period_end": "2025-09-23T17:00:00Z"
            }
        ],
        "exceptional_closings": [
            {
                "period_begin": "2025-05-23T08:00:00Z",
                "period_end": "2025-05-23T17:00:00Z"
```

```
                    }
                ]
            },
            "number_of_ports": 2,
            "ports": [
                {
                    "id": "82e095ee-499a-4ec4-939b-6f4cc9fd1ca2",
                    "connector_types": [
                        "IEC_62196_T1",
                        "NEMA_5_20"
                    ],
                    "charging_level": "AC LEVEL 2",
                    "charging_voltage_range": "400-800"
                    "power_delivery_kilowatts": 7.7,
                    "connected_trailer_access": false,
                    "status": "AVAILABLE",
                    "price_per_kWh": 0.33
                },
                {
                    "id": "d926c4a0-8763-4468-8bf0-d7ddd897e4fe",
                    "connector_types": [
                        "CHADEMO"
                    ],
                    "charging_level": "DC FAST",
                    "charging_voltage_range": "210-230"
                    "power_delivery_kilowatts": 120,
                    "connected_trailer_access": true,
                    "status": "CHARGING",
                    "price_per_kWh": 0.41
                }
            ],
            "payment_methods": [
                "CHIP_CARD_SUPPORT",
                "CONTACTLESS_CARD_SUPPORT",
                "CREDIT_CARD_PAYABLE",
                "DEBIT_CARD_PAYABLE"
            ],
            "pricing_structure": {
                "currency": "DOL",
                "components": [
                    {
                        "type": "FLAT",
                        "price": 1.00
                    },
                    {
                        "type": "TIME",
                        "price": 0.00,
                        "unit": "MINUTE"
                    },
                    {
                        "type": "PARKING_TIME",
                        "price": 0.25,
                        "unit": "MINUTE"
                    }
                ]
            }
        }
    ]
}
```

WHEN Data Are To Be Shared

Data Update Latency

Many of the data requirements in 23 CFR 680.116(c) define location and ownership and are unlikely to change over the operational life of the charging station. A change to any of the data (unless indicated otherwise) should be reflected in calls to the API within 24 hours of the event (for instance, if a charging station that was under construction becomes available for use at 8:00 a.m. EST on Jan. 4, 2024, that operational status of “open” should be reflected in calls to the API no later than 8:00 a.m. EST Jan. 5, 2024). Some of the data requirements, however, should be made available with a lower latency to ensure effective use. In addition to being returned via GET requests, this lower-latency data should also be sent to subscribed users via a PATCH request (see the section titled Scalability). It is recommended that the following data should be available with a lower latency of 15 seconds:

- **Real-time port status**: Live port status value should be reflected in GET requests to the API and sent to subscribers via PATCH request within 15 seconds of a change.
- **Pricing structure**: Changes to any component of the pricing structure should be reflected in calls to the API and sent to subscribers via PATCH request within 15 seconds of a change.
- **Real-time port price**: The live dynamic price component ($/kWh) should be reflected in calls to the API and sent to subscribers via PATCH request within 15 seconds of a change.

Response Time

Requests originating from IP addresses in North America should receive a response to GET requests from the API in less than 1,000 milliseconds, on average. Response time will result from a combination of factors and may vary from one request to the next, but an average response time less than 1 second should be easily achievable following modern development practices. It is also a response time that should not negatively impact end use cases.

Rate-Limiting Strategy

An appropriate rate-limiting strategy for charge point provider APIs can be found in the General Services Administration’s Developer Manual (General Services Administration 2024). A reasonable rate-limiting strategy will lower the costs of service maintenance by charge point providers, as well as provide a better end experience for API users by making the service more stable and secure. The API can reasonably enforce the following limits on API users:

- **Verified user key**: Users with a full key should be allowed to query and receive responses to at least 1,000 requests per hour. A higher allowance may be used if the charge point provider’s API infrastructure supports more than 1,000 requests per hour.

- **Demo key**: It is encouraged to allow users who want to survey the platform without registration to support independent GET requests (e.g. postman, curl, etc.) with an empty header or a hardcoded demo key. Requests to the platform using this approach may have a more stringent limit, but limit to no fewer than 30 requests per hour or 50 requests per 24 hours.
- **Pagination**: It is acceptable and encouraged that the APIs for networks with large numbers of assets implement a pagination schema to reduce the per-request burden to their API infrastructure.

It is recommended that each charge point provider only require a single regular API key per user, which will grant access to the entire network of charging stations with their associated data.

In addition to the data response, responses returned from the API should include headers that communicate rate limits to the client. Limits on the number of objects returned can be communicated using OCPI-compliant headers such as **X-Limit** and **X-Total-Count**. If the platform imposes limits on the number of requests, they can be additionally communicated using: **X-RateLimit-Limit** and **X-RateLimit-Remaining**. The value of **X-RateLimit-Limit** would always be the maximum allowed request per hour for the full key, and **X-RateLimit-Remaining** would be the number of requests left until the start of the next hourly request timer. These limits only apply to GET requests and do not apply to the traffic that is sent from the charge point provider's API via PATCH to registered URLs.

Availability

Following good practices for outage recovery, API services can be expected to be accessible along with the average uptime of the server resources on which they run. While comprehensive numbers on the average uptime of cloud services are largely industry secrets, cloud providers do provide uptime guarantees after which they will grant customers billing refunds. Common cloud services advertise 99.9%-plus uptime before issuing cloud cost refunds. Taking into account the refundable downtime average and including a buffer to handle scheduled downtime for maintenance actions, it is recommended that API downtime should not exceed 3.6 hours of total service downtime per calendar month, which translates to an uptime percentage of 99.5%.

Each of the major cloud providers provides auto-recovery products, including Amazon's Elastic Load Balancing, Google's Cloud Load Balancing, and Microsoft's Azure Load Balancer. As an alternative, the API can be deployed behind a service such as Docker Swarm or Kubernetes to automatically detect outages and recover services. These are just guiding examples, but some auto-recovery mechanism should be implemented for the API. Reference to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. government or any agency thereof, or its contractors or subcontractors.

WHO Can Access Data

API Users

The purpose of the charge point provider APIs is to provide necessary data for improving customer experience by allowing third-party software developers to use the station, port, and pricing data to help consumers make informed decisions about trip planning and when and where to charge their electric vehicles. As such, it is reasonable for charge point providers to plan on serving traffic to the base of third-party software developers and researchers who access charge point data directly, while traffic directly from the public at large is expected to be limited. This is verified by usage patterns of other similar data networks, such as Norway's NOBIL database (Godbolt 2024).

Charge point providers should make API documentation available online, including information on how to request token access for their API. Any third-party software developer should be able to request token access. Networks should use reasonable discretion when granting and revoking tokens.

Exact performance of any web service will depend on a variety of factors, including cloud or on-premises hardware specifications, programming language selection, efficiency of data queries, and network bandwidth. However, as a rule of thumb, the minimal (free-tier) hardware resources from a cloud provider with a typical server written with JavaScript Node performing simple single-layer database access should be able to consistently handle 100 requests per second. This approximate limit, combined with the normalized rate of requests specified in the Rate-Limiting Strategy section, should be able to handle consistent traffic from approximately 1,500 users at the same time.

Ultimately, the design of the service will need to respond to the traffic of verified users. Exact traffic is impossible to predict precisely, but a good target for performance would be to handle the normalized traffic from 10,000 total registered API users.

HOW Data Are To Be Shared

API Interface

Compliance With REST Principles

It is recommended that the API follow the design principles of a REST architecture. REST allows for flexibility in how the service is implemented (owners can choose the programming languages and architecture to serve the API) and also allows API owners and users to respond to changes in the structure of data (such as when a new charge connector is added to a station). The principles of REST (IBM 2023) are presented below:

1. **Uniform interface**: All API responses for the same resource should take the same form, regardless of origin.
2. **Client-server decoupling**: The client should not need anything to access the API's data other than the published URL and an authorization token. The server should not make any changes to the client other than returning the requested information. If the authorization token is obtained via credential exchange, or the client registers for PATCH updates, the client will also need to run a publicly accessible webserver.
3. **Statelessness**: No response from the API should require information or context that is not contained in the client's request.
4. **Cacheability**: In situations where it makes sense, data fields should be cacheable by the client, and cacheable fields should be indicated in the response.
5. **Layered system architecture**: The API should not assume or require that requests reach it directly from the client. There may be intermediate layers between the system initiating the request and the server.

Error Statuses

The API should implement and handle the error statuses outlined in Table 3, at a minimum. These error messages should be sufficient to instruct an API user to make necessary adjustments to when their requests fail from improper authentication configuration, limit overruns, and inappropriate request methods. API providers should also consider adding error messages sufficient to describe errors related to improper queries (e.g., when an API user requests station data with an invalid station identifier).

Table 3. Error Codes Returned by the API (General Services Administration 2024)

Error Code	HTTP Status Code	Description
API_KEY_MISSING	403	An API key was not supplied.
API_KEY_INVALID	403	An invalid API key was supplied. Double-check that the API key being passed in is valid, or sign up for an API key.
API_KEY_DISABLED	403	The API key supplied has been disabled by an administrator.
API_KEY_UNAUTHORIZED	403	The API key supplied is not authorized to access the given service.
API_KEY_UNVERIFIED	403	The API key supplied has not been verified yet. Please check your email to verify the API key.
HTTPS_REQUIRED	400	Requests to this API must be made over HTTPS. Ensure that the URL being used is over HTTPS.
OVER_RATE_LIMIT	429	The API key has exceeded the rate limits.
NOT_FOUND	404	An API could not be found at the given URL. Check your URL.

Nonfunctional Features

Some features of the API do not relate to the presentation of data. The following sections describe suggestions for the management and structure of the API not directly related to data access by API consumers.

API Platform Management

Principles for managing different API data regions and rolling them up into a single authentication portal can be found in the *API Umbrella Documentation* (National Renewable Energy Laboratory 2023). It is encouraged to review this document and implement its patterns, as they conform with the following nonfunctional feature suggestions. It is also possible to implement the installation of this platform as the API management solution, if needed.

Security

There are reasonable considerations that an API should make in the name of security. Security measures should be implemented to protect the secure data of charge point providers, prevent unnecessary recovery cost due to malicious cyberattacks, and preserve the functionality of the API for regular users. These measures include:

- **Cyber review**: Review the *NIST Cybersecurity Framework 2.0* (Raimondo and Locascio 2024).
- **Identity verification**: A request for an API token should be validated with a link sent to a valid email address.
- **Usage logging**: Requests to the platform should be logged along with information extractable from the request such as user IP address information.
- **Cybersecurity event recovery plan**: The platform serving the API should be rebuildable from scratch in the event of a cybersecurity event that necessitates pulling the platform down.
- **API token**: It is advised to restrict responses only to requests that carry a valid bearer token that is either registered in the platform and validated with an email link or marked as a demo key.

Usability

To ensure that the APIs are easy to find and access, all public end points, request formats, response formats (along with descriptions, such as what the value "PUBLIC" means as an "access_type" value), and steps to retrieve or exchange authentication tokens should be publicly documented. Documentation should be sufficiently complete that a user with sufficient technical proficiency to access API services through existing open-source tools (e.g. code containers, postman, curl, etc) could sign up for an API token and target fleetwide and station-specific data without any other instructions or communication outside of the documentation itself. The documentation should also be sufficient for the user to subscribe to future PATCH requests on their own long-running public servers.

Any necessary user-supplied inputs to the API, including header values, query parameters, path parameters, or POST body values and their corresponding formats, should be fully explained in the documentation. For instance, if the API requires a specific ID as a parameter to access a region of the API data, the documentation should provide a complete list of valid ID values that are accepted by the API, or else it should point the user to an API end point that will provide the exhaustive list of valid ID values.

Scalability

In alignment with OCPI specifications, this guide recommends a GET/PATCH architecture where users can register their own hosted services and receive event notifications on network assets to which they are subscribed. This pattern is expected to reduce computational burdens for both API providers and clients and lead to lower-latency data modeling (as compared with a pull implementation). Users may send GET requests to access the initial tree of charge point data, subscribe to assets of interest, and then receive PATCH requests to public URLs they upload to the API platform. All data, including the data labeled “low latency,” should also be available in response to GET requests, but together with the API key request limits and the preference for the subscription architecture, the recommendations in this document are designed to reduce the networking and computational burden of both charge point providers and API consumers, while remaining up to date with frequently changing data.

References

Alternative Fuels Data Center. 2024. “Data Included in the Alternative Fuel Stations Download.” Accessed April 10, 2024. afdc.energy.gov/data_download/alt_fuel_stations_format.

Ecma International. 2017. *Standard ECMA-404: The JSON Data Interchange Syntax, 2nd Edition*. ecma-international.org/publications-and-standards/standards/ecma-404/.

EVRoaming Foundation. 2021. *OCPI 2.2.1*. evroaming.org/app/uploads/2021/11/OCPI-2.2.1.pdf.

Federal Highway Administration. 2023. “National Electric Vehicle Infrastructure Standards and Requirements.” *Federal Register* 88: 12724–12757. www.federalregister.gov/documents/2023/02/28/2023-03500/national-electric-vehicle-infrastructure-standards-and-requirements.

General Services Administration. 2024. “Developer Manual.” Accessed April 23, 2024. api.data.gov/docs/developer-manual/.

Godbolt, L. 2024. “Interview: NOBIL Database Usage.” April 11, 2024.

IBM. 2023. “What is a REST API?” www.ibm.com/topics/rest-apis.

National Renewable Energy Laboratory. 2023. *API Umbrella Documentation: Release 0.15.1*. api-umbrella.readthedocs.io/_/downloads/en/latest/pdf/.

Raimondo, Gina, and Laurie Locascio. 2024. *NIST Cybersecurity Framework 2.0: Small Business Quick-Start Guide*. NIST SP 1300. doi.org/10.6028/nist.sp.1300.

U.S. Department of Transportation. 2023. “Charger Types and Speeds.” Last updated June 22, 2023. www.transportation.gov/rural/ev/toolkit/ev-basics/charging-speeds.