SURFACE VEHICLE RECOMMENDED PRACTICE

J2847™/3 NOV2023

Issued 2013-12
Revised 2023-11

Superseding J2847/3 MAR2021

Communication for Plug-in Vehicles as a Distributed Energy Source

RATIONALE

This update adds Sections 5, 6, and 7 to include the additional messages and stages for AC vehicle-to-grid (V2G) and AC vehicle-to-home (V2H). Updates to UL standards for these functions is also included.

TABLE OF CONTENTS

TO PLACE A DOCUMENT ORDER: **Tel: 877-606-7323 (inside USA and Canada)**
Tel: +1 724-776-4970 (outside USA)
Fax: 724-776-0790
Email: CustomerService@sae.org
SAE WEB ADDRESS: http://www.sae.org

1. SCOPE

This SAE Recommended Practice applies to a plug-in electric vehicle (PEV) which is equipped with an onboard inverter and communicates using IEEE 2030.5-2018. It is a supplement to the SEP2 standard, which supports the use cases defined by SAE J2836/3. It provides guidance for the use of the SEP2 distributed energy resource function set with a PEV. It also provides guidance for the use of the SEP2 flow reservation function set, when used for discharging. It is not intended to be a comprehensive guide to the use of SEP2 in a PEV.

Note that in this document, SEP2 is used interchangeably with IEEE 2030.5-2018.

1.1 Purpose

SAE J2836/3 defines two system architectures in which a PEV and the connected electric vehicle supply equipment (EVSE) together serve as a distributed energy resource (DER). The inverter could be located in the PEV in which case AC power would flow from the PEV through the EVSE to the electric power system. Alternatively, the inverter could be located in the EVSE, in which case DC power would flow between the PEV battery and the EVSE as needed by the inverter to perform the selected DER function.

The entity that contains the inverter should be the focus of any communication with the system that is controlling the DER devices for the DER application. This communication should be with the PEV in the case of an onboard inverter or with the EVSE in the case of an external inverter. While this recommended practice only applies to a PEV with an onboard inverter, it could optionally be followed by an EVSE manufacturer to guide SEP2 implementation for the case where the inverter is located in the EVSE. SAE J2836/3 provides guidance for updates required to SAE J2847/2 to provide a DER DC mode that will allow the inverter in an EVSE to use the PEV battery when operating together as a DER. This DER DC mode will be an inner loop control mode modeled on the internal PEV communication between the inverter and the battery management system for an onboard inverter.

This document has five specific objectives:

- Provide guidance for **WHAT** information needs be exchanged with the PEV using SEP2.

- Provide guidance for **HOW** to manage the actual exchange of information with the PEV using SEP2.

- Define **RULES** of engagement for the creation and use of information, the timing of information exchanges, and the resolution of conflicts by the PEV. The behavior of the PEV is governed by its own controls, displays, and embedded software. The purpose of SEP2 is only to move information to and from the PEV in support of the PEV functions. While SEP2 provides some recommended behavioral guidance, this document should be followed for the implementation of PEV functionality if there is a conflict.

- Identify **CHANGES** for the next version of SEP2 and IEC 61850 to better support the requirements of SAE J2836/3. In some cases, SEP2 supports SAE J2836/3 requirements, but IEC 61850 does not. In other cases, the reverse is true.

- Provide **EXAMPLES** of the use of SEP2 to perform selected V2G applications. There are too many possible V2G applications to create an exhaustive set of examples. However, a few examples can help guide the development of the embedded PEV software required to implement the protocol for the inverter functions.

2. REFERENCES

2.1 Applicable Documents

The following publications form a part of this specification to the extent specified herein. Unless otherwise indicated, the latest issue of SAE publications shall apply.

2.1.1 SAE Publications

Available from SAE International, 400 Commonwealth Drive, Warrendale, PA 15096-0001, Tel: 877-606-7323 (inside USA and Canada) or +1 724-776-4970 (outside USA), www.sae.org.

SAE J2836/3 Use Cases for Plug-in Vehicle Communication as a Distributed Energy Resource

2.1.2 IEEE Publications

Available from IEEE Operations Center, 445 and 501 Hoes Lane, Piscataway, NJ 08854-4141, Tel: 732-981-0060, www.ieee.org.

IEEE 1547:2003 Standard for Interconnecting Distributed Resources with Electric Power Systems

IEEE 1547:2018 Standard for Interconnection and Interoperability of Distributed Energy Resources with Associated Electric Power Systems Interfaces

IEEE 1547.1:2020 Standard Conformance Test Procedures for Equipment Interconnecting Distributed Energy Resources with Electric Power Systems and Associated Interfaces

IEEE 1547.9 Draft Guide for Using IEEE Std 1547™ for Interconnection of Energy Storage Distributed Energy Resources with Electric Power Systems

IEEE 2030.5-2018 IEEE Standard for Smart Energy Profile Application Protocol

IEEE 2030.7-2017 Standard for the Specification of Microgrid Controllers

IEEE 2030.8-2018 Standard for the Testing of Microgrid Controllers

IEEE 2030.9-2019 Recommended Practice for the Planning and Design of the Microgrid

2.2 Related Publications

The following publications are provided for information purposes only and are not a required part of this SAE Technical Report.

2.2.1 SAE Publications

Available from SAE International, 400 Commonwealth Drive, Warrendale, PA 15096-0001, Tel: 877-606-7323 (inside USA and Canada) or +1 724-776-4970 (outside USA), www.sae.org.

SAE J1772 SAE Electric Vehicle and Plug-in Hybrid Electric Vehicle Conductive Charge Coupler

SAE J2836/1 Use Cases for Communication Between Plug-in Vehicles and the Utility Grid

SAE J2836/2 Use Cases for Communication Between Plug-in Vehicles and Off-Board DC Charger

SAE J2836/3 Use Cases for Plug-in Vehicle Communication as a Distributed Energy Resource

SAE J2847/1 Communication for Smart Charging of Plug-in Electric Vehicles Using Smart Energy Profile 2.0

SAE J2847/2 Communication Between Plug-in Vehicles and Off-Board DC Chargers

SAE J2931/1 Digital Communications for Plug-in Electric Vehicles

SAE J2931/4 Broadband PLC Communication for Plug-in Electric Vehicles

SAE J3072 Interconnection Requirements for Onboard, Grid Support Inverter Systems

2.2.2 Electric Power Research Institute (EPRI) Publications

Available from EPRI, 3420 Hillview Avenue, Palo Alto, California 94304, www.epri.com.

EPRI 1026809 Common Functions for Smart InvertersIEC Publications

Available from IEC Central Office, 3, rue de Varembe, P.O. Box 131, CH-1211 Geneva 20, Switzerland, Tel: +41 22 919 02 11, www.iec.ch.

IEC 61850 Communication Networks and Systems for Power Utility Automation - Part 7-420, Basic Communication Structure - Distributed Energy Resources Logical Nodes

IEC 61850 Communication Networks and Systems for Power Utility Automation - Part 90-7, IEC 61850 Object Models for Photovoltaic, Storage, and Other DER Inverters
(Note that data models are now in IEC 61850-7-420 and are being deprecated in this technical report)

2.2.3 National Fire Protection Agency Publications

Available from NFPA, 1 Batterymarch Park, Quincy, MA 02169-7471, Tel: 617-770-3000, www.nfpa.org.

NFPA 70® National Electrical Code® (NEC®)

NEC® Article 625.27 Interactive Systems

NEC® Article 702 Optional Standby Systems

NEC® Article 705 Interconnected Electric Power Production Sources

2.2.4 UCA® International Users Group Publications

Available from UCA® International Users Group, 10604 Candler Falls Court, Raleigh, NC 27614, www.ucaiug.org.

UCAIug HAN SRS v2.0 UCAIug Home Area Network System Requirements Specification

2.2.5 UL Publications

Available from UL, 333 Pfingsten Road, Northbrook, IL 60062-2096, Tel: 847-272-8800, www.ul.com.

UL 1741 Standard for Inverters, Converters, Controllers and Interconnection System Equipment for Use with Distributed Energy Resources

2.2.6 ZigBee Alliance Publications

Available from ZigBee Alliance, 2400 Camino Ramon, Suite 375, San Ramon, CA 94583, www.zigbee.org.

ZB 09-5162 Smart Energy Profile 2.0 Marketing Requirements Document

ZB 09-5449 Smart Energy Profile 2.0 Technical Requirements Document

2.2.7 SunSpec Publications

Available from SunSpec Alliance, https://sunspec.org/ieee-2030-5-sae-j3072-ev-charging-profile-work-group/.

IEEE 2030.5 V2G-AC Profile IEEE 2030.5 V2G-AC Profile: Implementation Guide for SAE J3072, Version 1.0

3. DEFINITIONS

3.1 ADVANCED METERING INFRASTRUCTURE (AMI)

Advanced metering infrastructure (AMI) typically refers to the full measurement and collection system that includes meters at the customer site, communication networks between the customer and a service provider, such as an electric, gas, or water utility, and data reception and management systems that make the information available to the service provider.

3.2 AGGREGATOR

An individual PEV may not produce or consume enough power to be able to participate by itself in an application such as frequency regulation. An aggregator is an entity (which may not be a utility) that coordinates the power flow for a group of PEVs to allow them to meet the minimum power capacity required to participate in an application. The aggregator functions to, among other things, (a) schedule aggregated V2G services with a grid operator, utility, and other entities; (b) dispatch individual vehicles according to various factors including available V2G services and driver requirements; (c) aggregate response telemetry and relay to utility or grid operator; and (d) maintain a registry of vehicle identities and characteristics.

3.3 BEVSE

A bidirectional EVSE that can convert from AC to DC in one direction to serve as a battery charger, and then be capable of being reversed and convert from DC to AC in the other direction to serve as an inverter. This term is generally not used if the inverter is also capable of injecting or absorbing reactive power while charging or discharging.

3.4 BIDIRECTIONAL CONVERTER

See inverter definition (3.25).

3.5 CLIENT

In the standard client-server model, a client is the device or host that interacts with a server to obtain information related to a resource hosted by the server.

3.6 CONSISTENT OVERHEAD BYTE STUFFING (COBS)

Consistent overhead byte stuffing (COBS) is an algorithm for encoding data bytes that results in efficient, reliable, unambiguous packet framing regardless of packet content, thus making it easy for receiving applications to recover from malformed packets.

3.7 CONTROL SIGNAL

The UCAIug HAN SRS defines this as a structured message sent from an authorized party requesting operational state change of a device. Devices are expected to respond within the operation of their control systems and algorithms. This includes messages to DER devices for the purpose of controlling both active and reactive power.

3.8 COORDINATED UNIVERSAL TIME (UTC)

Coordinated Universal Time (UTC), once known as Greenwich Mean Time (GMT), is the reference time at the Greenwich meridian (London, UK). Time in SEP2 is defined as a signed 64-bit integer value representing the number of seconds from midnight January 1, 1970, in UTC, not counting leap second corrections to UTC (35 seconds through 2012).

3.9 DEMAND RESPONSE

A temporary change in electricity consumption by a demand resource (e.g., smart appliance, pool pump, PEV, etc.) in response to a control signal which is issued.

3.10 DISTRIBUTED ENERGY RESOURCE (DER)

Distributed energy resources (DERs) are small, modular distributed generation (DG) and storage technologies that provide electric capacity or energy where it is needed on the distribution grid. DG, which includes gensets, solar panels, and small wind turbines, only serve as a source of energy. Storage is a unique form of DER because, unlike pure DG, the unit can provide either energy or variable demand. Plug-in vehicles are DER storage systems.

3.11 EFFICIENT XML INTERCHANGE (EXI)

Efficient XML interchange (EXI) is a very compact, binary representation of the Extensible Markup Language (XML) information set. EXI format increases the processing speed of XML-based data, as well as reduces the memory usage.

3.12 ELECTRIC VEHICLE SUPPLY EQUIPMENT (EVSE)

Electric vehicle supply equipment (EVSE) is the generic term used to describe the device that is physically connected and provides energy to the vehicle. EVSEs may take several physical forms, and their logical function may likewise differ substantially. Physical forms include a mobile cordset used for 120 VAC charging, a fixed or wall-mounted 240 VAC charging station, or an off-board DC charger. In terms of logical function, any EVSE may or may not include one or more of the following: a “gateway” or physical layer bridge function to bridge PEV communications to the HAN, a device that communicates directly with the HAN itself, etc.

3.13 END DEVICE (EDEV)

If a device is not capable of hosting its own resources, it must place them on a host server where they can be discovered by other devices that need access to these resources. In SEP2, this set of client resources on a host server is considered to be its end device resources. These resources are uniquely associated with a specific client device.

3.14 END USE MEASUREMENT DEVICE (EUMD)

The end use measurement device (EUMD) is a revenue-grade meter responsible for directly measuring energy delivered to a specific end-device (e.g., PEV). This could be a dedicated utility meter if an EVSE is on a separate service. It could also be a sub-meter in the EVSE or PEV. The physical form, location and ownership of the EUMD may be unique for different applications. Revenue-grade means that the meter meets the requirements of the state entity that is responsible for ensuring the accuracy of the measurements, which will not necessarily be the public utility commission for a sub-meter that is not owned by a public utility.

3.15 ENERGY MANAGEMENT SYSTEM (EMS)

The term energy management system (EMS) is used in this document to describe a computer system that can communicate with a PEV or EVSE for the purpose of controlling the charging or discharging of the PEV battery. An EMS can exist at several tiers: customer premises, distribution level, or system level. These computer systems may go by other names, but the term EMS will be used generically in this document.

3.16 ENERGY SERVICES INTERFACE (ESI)

An energy serviecs interface (ESI) is a device on a HAN which enables secure communications between authorized parties (e.g., utility, service provider, aggregator, home EMS) and all commissioned HAN devices that are registered to it. HAN architecture allows for more than one ESI in a consumer premises. Each ESI creates an independent logical power distribution network within the premises, each with its own security.

3.17 ENROLLMENT

The process by which a consumer enrolls a HAN device in a service provider's program (e.g., demand response, energy management, pre-pay, PEV programs, distributed generation programs, pricing, messaging, etc.) and gives certain rights to the service provider to communicate with their HAN device.

3.18 EXTENSIBLE MARKUP LANGUAGE (XML)

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The design goals of XML emphasize simplicity, generality, and usability over the internet. It is a textual data format with strong support via Unicode. Although the design of XML focuses on documents, it is widely used for the representation of arbitrary data structures; for example, in IEEE 2030.5-2018.

3.19 FORWARD POWER FLOW (FPF)

Forward power flow (FPF) means the direction of energy for charging a vehicle. In SEP2, a setpoint command for forward (charging) power flow for a DER device has a negative value.

3.20 FUNCTION SET

A function set is a logical grouping of resources in the SEP2 object model (e.g., metering, demand response and load control, flow reservation, DER). All function sets do not have to be implemented in every device and the requirements associated with a given function set may be mandatory or optional.

3.21 HOME AREA NETWORK (HAN)

In this document, a home area network (HAN) is an energy related network, contained within a premises used for communicating with devices within the premises. HANs do not necessarily require connectivity outside the premises, but may be connected to one or more external communication networks using gateways, bridges and interfaces.

3.22 HOST

This is the representation of a device in its application context, typically represented by an IP address or domain name.

3.23 HYPERTEXT TRANSFER PROTOCOL (HTTP)

Hypertext transfer protocol (HTTP) is an application layer protocol designed within the framework of the Internet Protocol Suite. Its definition presumes an underlying and reliable transport layer protocol. HTTP resources are identified and located on the network by Uniform Resource Identifiers (URI). HTTP is the foundation of data communication for the World Wide Web. HTTP functions as a request-response protocol in the client-server computing model. The client submits an HTTP request message to the server which returns a response message to the client. The response contains completion status information about the request and may also contain requested content in its message body.

3.24 INADVERTENT EXPORT

Under certain operating conditions, the customer may choose to completely offset their facility load by using the EV as a generation system which are optimally sized to meet their peak demand with load following functionality on the Generator controls to ensure conditional export of electrical power from the generating facility to distribution provider's distribution or transmission system. In situations where the loading changes rapidly and/or the generator cannot ramp down quickly enough, the generating system may need to export small amounts of power for limited duration. The event of exporting uncompensated power for a short time is referred to as inadvertent export.

3.25 INVERTER

AC power is generated from a DC source, such as a traction battery, using a device called an inverter. A pure inverter cannot convert AC to DC for charging a battery; this is done by a charger or a forward converter. Bidirectional converter is the term used for a device that can convert from AC to DC in one direction to serve as a battery charger and also be capable of being reversed and convert from DC to AC in the other direction to serve as an inverter. A grid-tied power converter can be designed to either lead or lag its sourced current relative to the grid voltage during either charging or inverter operation. This is called a four quadrant converter. It is common practice to refer to the power convertor of a storage DER device as “inverter,” even though this is not precisely correct. An onboard inverter is included within the vehicle and an offboard inverter is located off the vehicle, in the EVSE.

3.26 PLUG-IN ELECTRIC VEHICLE (PEV)

This is the generic term used to describe any vehicle that plugs in to receive electrical energy. This includes many different classifications of vehicles, such as BEVs (battery electric), PHEVs (plug-in hybrid electric), E-REVs (extended-range electric), and so on.

3.27 POWERLINE COMMUNICATIONS (PLC)

Powerline (carrier) communications (PLC) refers broadly to the group of communications technologies in which a modulated carrier is transmitted over AC power circuits. The same communications technologies and designs may sometimes also be applied to circuits that do not carry AC current.

3.28 REPRESENTATIONAL STATE TRANSFER (REST)

Representational state transfer (REST) is a style of software architecture for distributed systems such as the World Wide Web. SEP2 uses REST architecture. REST architectures consist of clients and servers. Clients initiate requests to servers; servers process requests and return appropriate responses. Requests and responses are built around the transfer of representations of resources. A representation of a resource is typically a document that captures the current or intended state of a resource.

3.29 RESOURCE

URI addressable object that is manipulated via the REST uniform interface

3.30 RESOURCE DISCOVERY

This is a process whereby clients identify resources being served on the network. Clients issue a request to all devices on the network requesting resource(s) of interest. Servers hosting the requested resource(s) respond with information necessary to access the server and its resource(s).

3.31 REVERSE POWER FLOW (RPF)

Reverse power flow (RPF) means the direction of energy for discharging a vehicle. In SEP2, a setpoint command for reverse (discharging) power flow for a DER device has a positive value.

3.32 SCHEMA

A data structure definition for XML data used to describe the structure, content, and to some extent, the semantics of XML documents. The IEEE 2030.5-2018 schema is described by a collection of XML schema definitions (XSDs), which are written in XML.

3.33 SELF DEVICE (SDEV)

If a device is a server and hosts its own resources it is considered to be a self device (SDEV) in SEP2. A complex device, such as a stationary storage DER unit, might provide its own discoverable server to host its resources. Because the connection of a PEV to the network is intermittent a PEV would most likely only act as a client and host its end device resources on an ESI or another server.

3.34 SERVER

In the standard client-server model, a server is the device or host that holds a resource and exposes representations of that resource.

3.35 SMART ENERGY PROFILE 2.0 (SEP2)

The ZigBee Alliance and HomePlug Powerline Alliance created the Smart Energy Profile 2.0 (SEP2), which defines a common information model, a common set of services, an application protocol implementation, and a set of behaviors for enabling smart energy applications while placing requirements on the underlying network stack, link layer, and security. When implemented correctly, the Smart Energy Profile will allow interoperability between different smart energy devices.

In 2014, IEEE took over governance of the standard from the ZigBee Alliance. Smart Energy Profile 2.0 is now known as IEEE 2030.5. The current version is IEEE 2030.5-2018.

3.36 UNIVERSAL ASYNCHRONOUS RECEIVER-TRANSMITTER (UART)

Universal asynchronous receiver-transmitter (UART) is a hardware communication protocol that uses asynchronous serial communication with configurable speed. Asynchronous means there is no clock signal to synchronize the output bits from the transmitting device going to the receiving end

3.37 UNIFORM RESOURCE IDENTIFIER (URI)

Uniform resource indentifier (URI) is a string of characters used to identify a name or a resource. Such identification enables interaction with representations of the resource over a network (typically the World Wide Web) using specific protocols. URIs can be classified as locators (URLs), as names (URNs), or as both. A uniform resource name (URN) defines an item's identity and a uniform resource locator (URL) provides a method for finding it.

3.38 VEHICLE-TO-GRID (V2G)

When vehicle power is fed into the bulk electric grid or a microgrid, we refer to it as “vehicle-to-grid” power, or V2G. V2G is the only mode that allows a vehicle to return power to a home, business, or charge station which is actively connected to the grid. This could be at home or remote locations and initiated for contract driven ancillary services (regulation, peak shaving, reserves, etc.). A PEV in V2G operation is considered by utilities to be a DER.

3.39 WEB-APPLICATION DESCRIPTION LANGUAGE (WADL)

Web-application description language (WADL) is a machine-readable XML description of HTTP-based web applications (typically REST web services). It models the resources provided by a service and the relationships between them. It is intended to simplify the reuse of web services that are based on the existing HTTP architecture of the Web. It is platform and language independent and aims to promote reuse of applications beyond the basic use in a web browser.

4. TECHNICAL REQUIREMENTS

This major section provides recommended practices for the use of IEEE 2030.5-2018 for communication by a PEV that is equipped with an onboard inverter capable of interconnecting with an electric power system as a DER. This document does not require a PEV to be equipped with an onboard inverter, nor does it require that SEP2 be used for communication; in either case, this document is not applicable to such a PEV.

In this document, the term “inverter” is used two ways, depending on context. The requirements for interconnecting a DER to the grid only apply to the production of power (i.e., discharging a storage DER) and in this context the relevant power conversion functionality in the PEV is that of a pure inverter device. But a storage DER, such as a PEV, can also absorb power from the grid and the power conversion electronics would properly be called a battery charger or a forward converter. It is possible that a PEV could have separate inverter and charger devices, although an integrated bidirectional or four-quadrant power converter would more likely be more cost effective. For storage DER, the direct control of active power from maximum production to maximum consumption is considered to be an inverter function even though this is not precisely correct. In this document, the term inverter is used in the context of DER functions being inverter functions.

This document assumes that a PEV meets the following requirements:

- The PEV has an onboard, utility-interactive inverter. The PEV and EVSE to which it is connected using the SAE J1772 coupler shall meet the requirements of NEC® Article 625.27 for an electric power production source. This article calls out NEC® Article 705, which defines requirements for a “utility-interactive inverter.”

- The inverter shall provide the functionality to fully support SAE J2836/3 use case U6 (Basic Distributed Energy Resource). The inverter shall be capable of seamless transition between forward and reverse active power flow in response to a control signal. The inverter is not required to be capable of producing or absorbing reactive power to support U6.

- The inverter may optionally provide the functionality to support one or more of the modes defined by SAE J2836/3 use case U7 (Advanced Distributed Energy Resource). Some of these modes require an inverter to be capable of producing or absorbing reactive power. Other modes require autonomous operation, some of which may also require reactive power control. It is possible that a specific utility could require a DER to be capable of supporting one or more of these advanced modes as a condition of their interconnection agreement.

- The inverter shall provide the functionality to fully support SAE J2836/1 use case U5 (Optimized Energy Transfer). The inverter shall be capable of varying the discharge rate in response to a control signal when the PEV is engaged in discharging. Seamless transition between charging and discharging is not required.

- The PEV communication system shall comply with the SEP2 standard, except as may be constrained by this document. This includes all of the supporting protocols defined by the SEP2 standard. The communication system shall also comply with SAE J2931/1; if there is a conflict with SEP2 requirements, SEP2 shall govern.

- The PEV shall implement all mandatory resources of the SEP2 model for the power status, flow reservation, and distributed energy resource function sets and all of those optional resources that are needed to support the PEV inverter functional capability.

- SEP2 is data link agnostic. Examples in this document are based on Powerline Communications (PLC) on the SAE J1772 control pilot using HomePlug Green PHY as defined by SAE J2931/4. A PEV could use another data link and this document would still apply.

This document is a supplement to the SEP2 standard. It provides guidance for the use of the SEP2 distributed energy resource function set with a PEV. It also provides guidance for the use of the SEP2 flow reservation function set, when used for discharging. It is not intended to be a comprehensive guide to the use of SEP2 in a PEV. There are many SEP2 function sets that could be implemented in a PEV, some of which are required to support a PEV operating as a DER. For example, an application that wishes to read the current flow from the inverter might need to use the metering function set to be able to retrieve that information from a PEV. This document does not describe the use of the metering function set. A DER controller for a specific V2G application may need to engage with a PEV using other SEP2 function sets beyond the DER function set. The SEP2 standard describes the full communication capability.

The SEP2 standard focuses on the communication of information, but it does not define the creation of information by a PEV or the use of control signals or other information by a PEV. A primary focus of this document is to provide guidance for the creation and use of information which is done by the embedded operational software of the PEV. Some information may require data entry or other out of band processing during initial set up for a newly enrolled V2G application or for a specific session. This is beyond the scope of SEP2, but directly relevant to the ability to perform the DER function.

A brief overview of each subsection is provided below:

Section 4.1 Provides a brief overview of the basic system architecture for a PEV engaging as a DER device with a DER controller.

Section 4.2 Provides a brief overview of the communication stack associated with SEP2 and the relationship of each layer to SAE documents SAE J2836/3, SAE J2847/3, SAE J2931/1, and SAE J2931/4.

Section 4.3 Defines the key documents of SEP2 and the SEP2 function sets that are required to support SAE J2836/1 use case U5 (Optimized Energy Transfer) and SAE J2836/3 use cases U6 (Basic DER) and U7 (Advanced DER).

Section 4.4 Provides a very brief layman's overview of some of the concepts that will be needed to appreciate the guidance provided in this document for how to apply SEP2.

Section 4.5 Reviews two approaches for an EVSE to handle the PLC communication with the PEV. In one approach, the EVSE only "bridges" the PEV to a SEP2 HAN by converting, for example, PLC to ethernet. In the other approach, the EVSE serves as a gateway and performs protocol translation for the PEV.

Section 4.6 Presents some gaps between SEP2 and the IEC 61850-7-420 object model. The SEP2 model conforms to SAE J2836/3 U6 requirements and EPRI and IEC are working to include the additional information required by U6.

Section 4.7 Provides guidance for the use of the SEP2 flow reservation function set for discharging as defined by the SAE J2836/1 use case U5.

Section 4.8 Provides guidance for creating the PEV resource information required by the DER function set and the power status function set to support SAE J2836/3 use cases U6 and U7.

Section 4.9 Describes the SEP2 concept of a DER Program which is a resource created by a DER controller that could apply to may DER devices. A DER Program could apply to only one PEV or a group of PEVs could be enrolled in a program.

Section 4.10 Defines the structure of a DER Control object. This is a structured control signal that a DER controller uses to manage the PEV inverter as required by the V2G application.

Section 4.11 Defines rules and guidelines for event conflicts. A DER control signal is defined as an event in SEP2, but there are other events, such as a demand response event.

Section 4.12 Provides guidance for using the autonomous curve functions of the DER function set.

Section 4.13 Provides guidance for using the ride through function of the DER function set.

Section 4.14 Provides a brief overview of two V2G application examples. The details for each example are provided in a separate appendix.

4.1 Communication System Architecture

The basic system architecture in which a PEV can participate as a DER is shown in Figure 1. The PEV has an onboard inverter and uses IEEE 2030.5-2018 for communication. The purpose of the communication system is to allow a DER controller to receive state information from the PEV and to allow the DER controller to provide timely control signals to the PEV. This end-to-end information exchange is illustrated by the dotted lines connecting the PEV with the DER controller.

If the DER controller does not use SEP2, protocol translation will be required by some network device located between the DER controller and the PEV. This translation will be easier if the DER controller conforms to the IEC 61850 object model for a DER inverter. The DER function set of SEP2 is based on IEC 61850-7-420. SEP2 supports the use cases defined by SAE J2836/3 which was also based on the IEC 61850 DER model. The figure shows the PEV communicating with the DER controller by way of the EVSE using powerline communications (PLC) on the SAE J1772 control pilot. The specific PLC implementation would be HomePlug Green PHY as governed by SAE J2931/4. This path to the network will be used for examples in this document, although SEP2 can operate using other physical layers. For example, it could be possible for a PEV that is equipped with a wireless link to bypass the EVSE and directly engage with a premises network that is based on SEP2.

Any device which can discharge power into an electric power system that is connected to the utility grid must be approved by the local utility to interconnect to the electric power system. This is true even if no net energy is delivered from the premises. The interconnection agreement with the utility will be based on a defined point of common coupling (PCC), which will generally be the utility meter at the premises. A PEV should not be permitted to discharge unless it is authorized to do so by the interconnection agreement that applies to the specific premises. Because the EVSE can be associated with the specific PCC covered by the interconnection agreement, it is essential to associate the PEV with the EVSE to which it is connected. This association with the EVSE can be less complex if the PEV only communicates with or through the EVSE. It is a more complex problem if the PEV uses an alternate communication path around the EVSE. This is only part of the problem because even though the connection path through the EVSE is confirmed, discharging can only be authorized by the PEV if the specific PEV model is approved by the interconnection agreement. For a residential application, this could be a manual confirmation by the driver using vehicle controls and displays. The authorization to discharge process is much more complex for a public charging location.

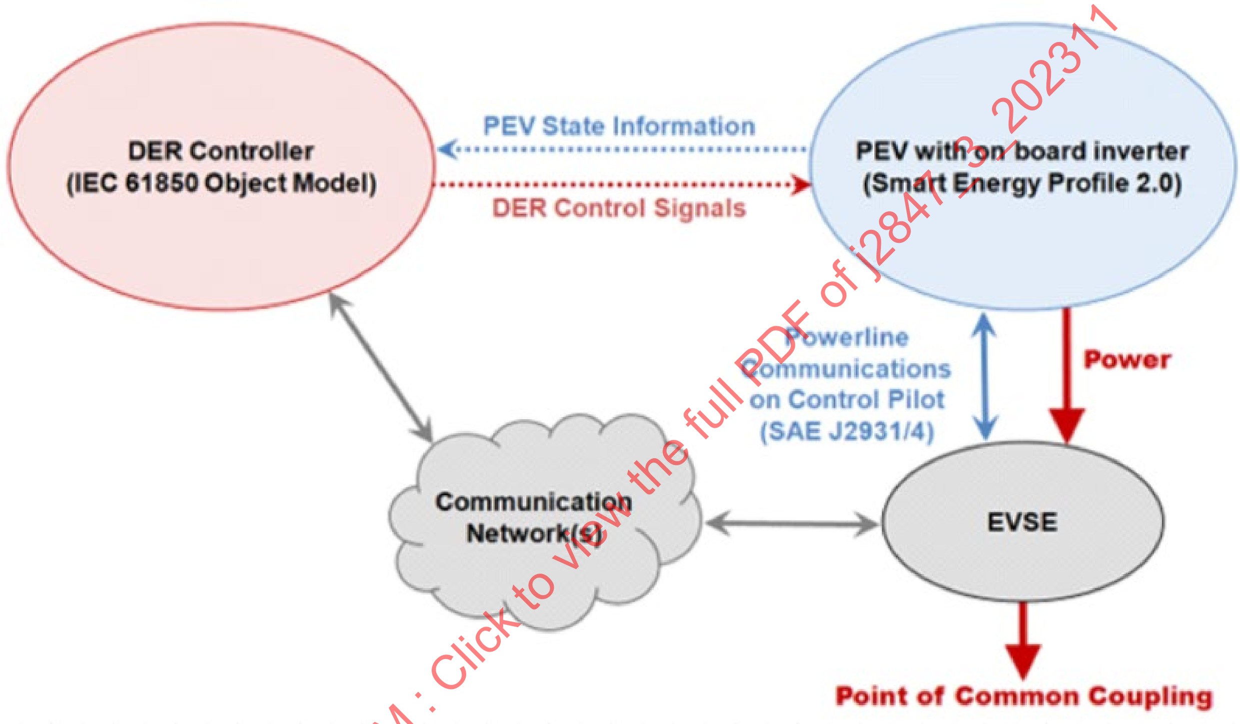


Figure 1 - Basic system architecture

4.2 Communication Stack

IEEE 2030.5-2018 is based on the Internet Protocol Suite which is the set of communications protocols used for the internet and similar networks, and is generally the most popular protocol stack for wide area networks. It is commonly known as TCP/IP, because of its most important protocols: Transmission Control Protocol (TCP) and Internet Protocol (IP). An alternative model is the open systems interconnection (OSI) model which characterizes and standardizes the functions of a communications system in terms of seven abstraction layers. Figure 2 shows the seven OSI layers as grey bars and overlays the four TCP/IP layers as blue rectangles. The white rectangles show some of the protocols or standards used at each layer with SEP2.

The application layer of both the TCP/IP and OSI models is closest to the embedded DER software which is indicated by the light blue bar. The application software of the HAN device is outside the scope of the OSI and TCP/IP models. The green bar represents the data and functional model for the inverter that guides the development of the embedded PEV software.

The red rectangles show the relationship of SAE documents to the different layers. SAE J2931/4 governs the use of high bandwidth powerline communication between the PEV and EVSE on the SAE J1772 control pilot. SAE J2931/1 deals with the protocols used for the internet and transport layers and some of the protocols used in the application layers. There is overlap with SAE J2847/3 for the application layer. For example, the need to have HTTP as one of the application layer protocols used by the PEV is governed by SAE J2931/1. The specific HTTP message structures needed to send and receive DER control signals are covered by SAE J2847/3. SAE J2836/3 extends to the application level because it establishes requirements for the DER model which in turn establishes requirements for the XML schema definition (XSD).

SAE J2931/7 establishes the security requirements for digital communication between the PEV, the EVSE, and other entities in the overall system. Consequently, security will not be addressed in this document.

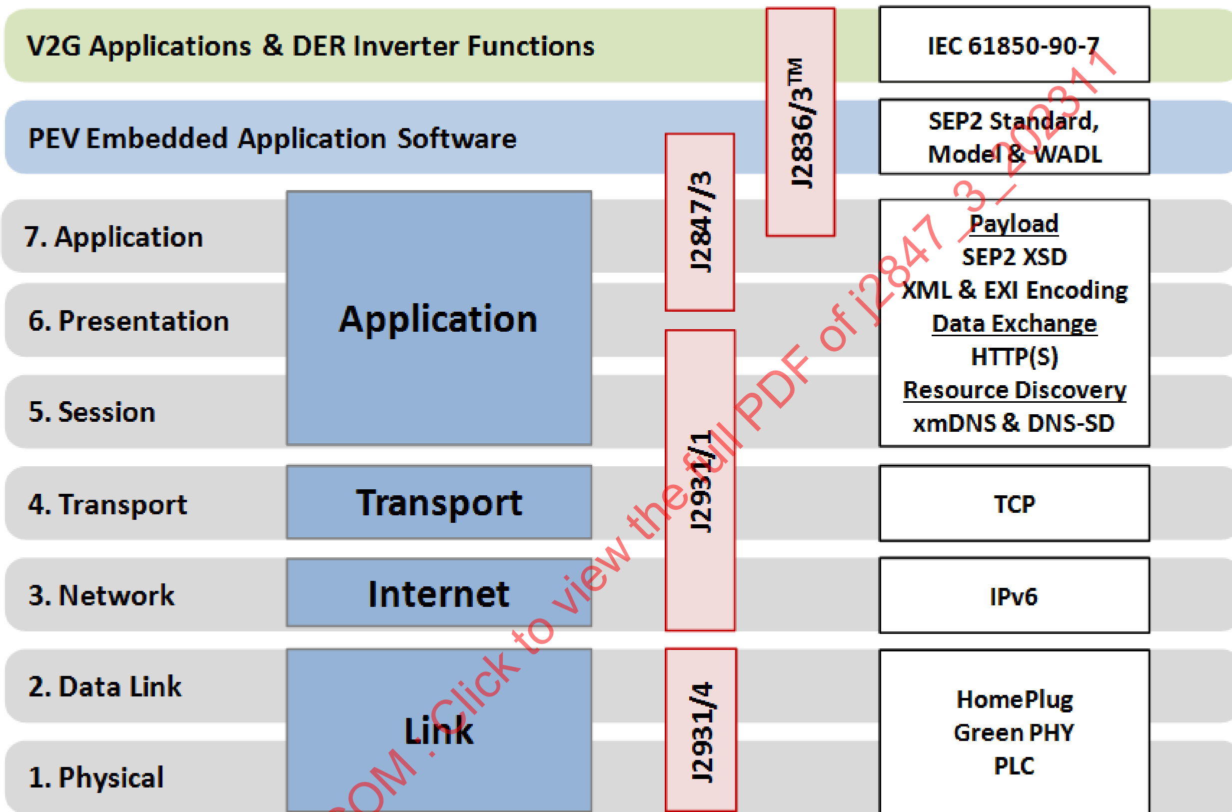


Figure 2 - PEV communication stack

4.3 IEEE 2030.5-2018 - Standard for Smart Energy Profile Application

IEEE 2030.5-2018 is designed to meet the requirements defined by the SEP2 Marketing Requirements Document and the SEP2 Technical Requirements Document. These two documents are based on the UCAIug Home Area Network System Requirements Specification.

The SEP2 Application Protocol is at the application layer of a TCP/IP stack with HTTP(S)/1.1 serving as the application data exchange protocol. Internet Protocol version IPv4 or IPv6 can be used. SEP2 is intended to enable communications that are link layer agnostic.

SEP2 is based on Representational State Transfer (REST). An important concept in REST is the existence of resources (sources of specific information), each of which is referenced with a global identifier (e.g., URI in HTTP). A server is a device that hosts a resource, and a client is a device that obtains, extends, updates, or deletes representations of that resource. A representation of a resource is typically an XML document that captures the current or intended state of a resource. Clients initiate requests to servers via a standardized interface (e.g., HTTP) and servers process requests and return appropriate responses.

In SEP2, the HTTP documents can be encoded using either Extensible Markup Language (XML) or Efficient XML Interchange (EXI). EXI is a compact alternative encoding of XML documents. A server must be able to send and receive messages either encoded as a plain XML string or encoded as an EXI stream. A client can implement both, but is only required to implement one. The rich variety of interactions on a HAN can make it difficult to define “client” and “server.” A specific HAN device could serve as a client and as a server for different functions. Devices that can act as both a client and a server would need to support both encoding methods.

The following documents comprise the definition of the SEP2 Application Protocol and all SEP2 devices will be required to maintain compliance to these documents:

- IEEE 2030.5-2018: Standard for Smart Energy Profile Application Protocol
- IEEE 2030.5-2018: Web-Application Description Language (WADL)
- IEEE 2030.5-2018: XML Schema Definition (XSD)

SEP2 uses a Web-Application Description Language (WADL) to define the recommended URI structures and the use of the HTTP methods of GET, PUT, POST, and DELETE. The default mechanism for obtaining a resource representation is a pull mechanism, implemented with GET. The use of a subscription mechanism for retrieving a resource representation is also optionally supported in many instances, where convenient and appropriate. SEP2 devices must conform to the resource specifications contained in the WADL. An Informative textual extract of the SEP2 WADL is contained in the Application Protocol Standard for convenience.

The object model of the SEP2 Application Profile is based on the IEC 61968 common information model (CIM), mapping directly where possible, and using subsets and extensions where needed. For example, the object model of the SEP2 DER function is based on IEC 61850-7-420 and not the CIM. The object model is described by a collection of XML schema definitions (XSDs) which are written in XML. The SEP2 XSD contains the definitions of the resources, attributes, and elements as well as their textual descriptions. An informative textual extract of the SEP2 XSD is contained in the Application Protocol Standard for convenience.

The Application Protocol contains three collections of function sets: support, common, and smart energy. The support function sets provide operational information to the end devices of an SEP2 network or provide those end devices with services to manage and support their operation. Common function sets provide general purpose, non-domain-specific functionality. Smart energy function sets are specific to the domain of smart energy. Each function set is fully described in the Application Protocol Standard. The WADL and the XSD follow a similar organization by function set.

A function set is the set of behaviors necessary to support a given business requirement, such as the ability of a PEV to serve as a DER device. All function sets do not have to be implemented in every device and the requirements associated with a given function set may be mandatory or optional. Devices implementing a particular function set are required to implement all mandatory requirements for that function set corresponding to the device’s role. The expression [0..1] is used in the data model (XSD) to indicate that a particular object or attribute is optional for that function set.

Table 1 shows which function sets are used by the Basic DER (U6) and Advanced DER (U7) use cases from SAE J2836/3. The Optimized Energy Transfer (U5) use case from SAE J2836/1 is also shown because it can be used for reverse power flow in some V2G applications. BASIS designates the core function(s) for the use case. RQD designates a function that is required to support the core DER or flow reservation functions. OPT designates a function which could be optionally used specifically for the purpose of supporting a core function. For example, subscription could be used with the DER function as an optional way of getting control signals. A blank space does not indicate functions that would not be used in PEVs. Many are used in PEVs but they are not primarily needed to support these use cases. For example, metering could be used to support payment for participating as a DER in an application, but it is not essential to the basic operation of the DER function. This document will primarily focus on providing guidance for the BASIS function sets.

Table 1 - Resources and function sets

Use Cases	SAE J2836/1 U5	SAE J2836/3 U6/U7
Support Resources		
Device Capabilities Function Set	RQD	RQD
Self Device Resource		OPT
End Device Resource	RQD	RQD
Function Set Assignments	RQD	RQD
Subscription/Notification Mechanism		OPT
Response	RQD	RQD
Common Resources		
Time Function Set	RQD	RQD
Device Information Function Set	RQD	RQD
Power Status	OPT	**BASIS**
Network Status		
Log Event List		
Configuration Resource		
File Download Function Set		
Smart Energy Resources		
Common Functionality	RQD	RQD
Demand Response and Load Control		
Metering Function Set		
Pricing Function Set		
Messaging Function Set		
Billing Function Set		
Prepayment Function Set		
Energy Flow Reservation Function Set	**BASIS**	
Distributed Energy Resources Function Set		**BASIS**
Metering Mirror		

4.4 Some Basics about SEP2 for the Non-Expert

The purpose of this section and its subsections is to provide a very brief layman's overview of some of the concepts that will be needed to appreciate the guidance provided in this document for how to apply the SEP2 protocol to a PEV engaging as a DER in a V2G application. Please use the actual SEP2 standard as the authoritative reference. This section is not intended for those people that are highly skilled with SEP2 and its associated processes and protocols.

4.4.1 Clients, Servers, and Resources

SEP2 is based on REST, in which a client polls a server using the HTTP GET method to obtain a representation of the current state of a resource. A device must be capable of acting as an SEP2 server for it to directly share its resources with other SEP2 network devices. Otherwise it must place (PUT) copies of the resources that it wishes to share on an end-device server that can host them and make them available to GET requests from other authorized network devices. In SEP2, the distinction between a server and a client arises dependent on whether a device exposes a resource (server) or interacts with the resource (client). A device which is a server for its own resources would be a client when requesting resources hosted by another server. A device which is not capable of hosting resources can only act as a client.

Figure 3 shows an example where one DER device is a server (DER-A) and another is only a client (DER-B). Blue designates a client behavior and red designates a server behavior. In this example, each DER device creates a resource "ABC" and the DER controller creates a resource "EFG." For DER-A, the controller, as a client, uses an HTTP GET method to request resource "ABC" from DER-A. The DER-A server responds by providing an XML or EXI message containing a representation of resource "ABC." In a similar manner, DER-A can GET "EFG" from the controller. It is more complex for the case of DER-B which is not directly visible to the controller. The controller interacts with an end device representation of DER-B resources on an authorized end device server (ESI). The controller uses the GET method with the end device server to request "ABC" and the server responds with the XML for "ABC." This requires DER-B to place a copy of "ABC" on the host server so it can be discovered and retrieved by the controller. It uses the PUT method to maintain the "ABC" resource on the host server. DER-B can directly discover the "EFG" resource on the controller server just as with DER-A. If the controller could host DER-B resources, DER-B could PUT "ABC" directly into the controller and the local ESI server may not be needed.

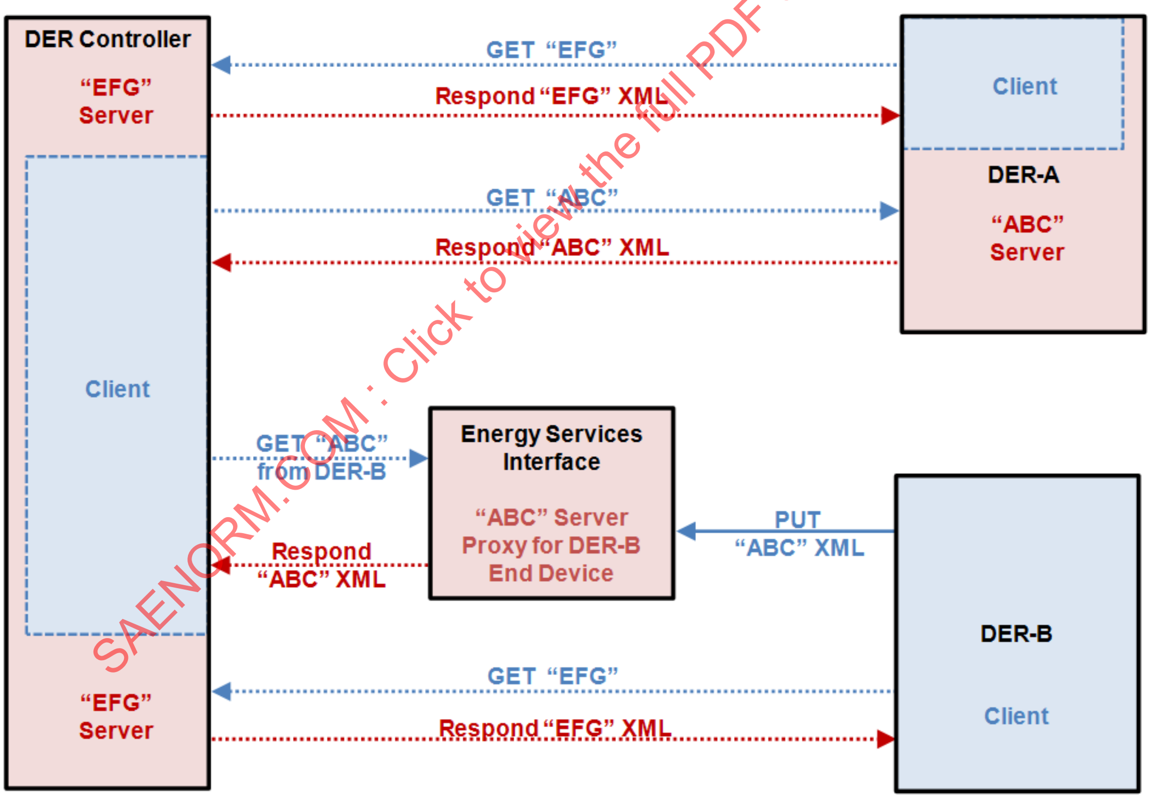


Figure 3 - DER device as client or server

4.4.2 Extensible Markup Language (XML)

The information payload in SEP2 messages can be encoded using either XML or EXI. XML is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. EXI is a compact alternative encoding of XML documents which allows XML-based messages to be used and processed on a binary level. Thus, the EXI format increases the processing speed of XML-based data as well as reduces the memory usage. A server must be able to send and receive messages either encoded as a plain XML string or encoded as an EXI stream. A client can implement both, but is only required to implement one. Because of its human-readable structure, only XML will be discussed in this document. It is hard to describe anything at the binary level.

The characters making up an XML document are divided into markup and content. A markup construct that begins with “<” and ends with “>” is called a tag. There are three types of tags: start-tag (e.g., <xyz>), end-tag (e.g., </xyz>), and empty-element tag (e.g., <xyz/>).

An element is a logical document component which either begins with a start-tag and ends with a matching end-tag or only consists of an empty-element tag. The characters between the start-tag and end-tag, if any, are the element's content (e.g., <xyz>content</xyz>). The content may consist of other elements, which are called child elements. In SEP2, there can be several levels of child elements. This is often shown in an example of an XML document as a nested hierarchy to improve readability. The actual transmission is a tightly packed string of characters without spaces.

Attribute is the term used for a markup construct consisting of a name/value pair that exists within a start-tag or empty-element tag. This is the source of some confusion because the SEP2 model uses the term attribute as a property of a defined object and this named attribute then appears as an XML element. The SEP2 model uses the expression <<XSDattribute>> to define information that is intended to appear as an actual XML attribute. This will be discussed in more detail later.

A simple example of an XML document is shown below:

```
HTTP/1.1 200 OK
Content-Type: application/sep+xml
Content-Length:{contentLength}

<DERSettings href="/edev/1/der/1/derg" xmlns="urn:ieee:std:2030.5:ns">
     …
    <setMinPFOverExcited>
        <!-- PF = 0.995  -->
        <displacement>995</displacement>
        <multiplier>-3</multiplier>
    </setMinPFOverExcited>
    …
</DERSettings>
```


This is part of an XML payload that would come if a client requested a resource from a server using the HTTP/1.1 GET method. The server returns the response “HTTP/1.1 200 OK” followed by the XML message. The Content-Type and Content-Length are considered the “XML declaration” and tell the client device that the information is encoded in SEP2 and XML and the length.

DERSettings is the name of a DER resource object in SEP2. The start-tag has the element name DERSettings and includes two attributes within the tag. The “href” attribute provides the URI for this resource on the server. This was used in the GET. The “xmlns” attribute provides the link to the most recent XML namespace.

One of the attributes of the DERSettings object is called setMinPFOverExcited and it defines the minimum power factor for the DER device when injecting reactive power (over-excited). This is only one of several device settings returned in this XML message. This object attribute name appears as an XML element name. The minimum over-excited power factor is 0.995 for this device. SEP2 does not support floating point data types (although they are supported by generic XML schema definitions). This requires an integer value of 995 to be provided along with a power of ten multiplier value of -3. Comments start with "<!--" and end with "-->" and an example is also shown. It is not likely that comments would be embedded in machine to machine exchanges, but they are sometimes added in XML examples.

4.4.3 Hypertext Transport Protocol (HTTP) Methods

SEP2 is based on REST. An important concept in REST is the existence of resources (sources of specific information), each of which is referenced with a global identifier (e.g., URI in HTTP). Requests and responses are built around the transfer of representations of resources. A representation of a resource is typically a XML document that captures the current or intended state of a resource.

The communication of information using SEP2 is much like the process that a web browser uses to request and then display a web page. The browser requests the content of the webpage from the host server by using the HTTP GET method with the URI address of the specific page that is desired. The "resource" is located by its URI address. The server responds to the request by providing a "payload" which would consist of an HTML document. The browser uses the HTML document to create the display of the web page. The document could contain "links" to other resources. The links are just URI addresses to other pages (resources). A SEP2 payload is an XML (or EXI) document rather than an HTML document, but the fundamental process is the same.

Clients use the HTTP/1.1 methods of GET, POST, PUT, and DELETE to engage with servers. A server is a device that hosts a resource, and a client is a device that obtains, extends, updates, or deletes representations of that resource. A server device that requests a resource from another server is acting as a client for that resource.

A resource is requested from a server by using the GET method with a URI providing the address of the resource as shown below:

```
GET /edev/1/der/1/derg HTTP/1.1
Host: {hostname}
```

The end device server hosting the DER resources responds with:

```
HTTP/1.1 200 OK
Content-Type: application/sep+xml

<DERSettings href="/edev/1/der/1/derg" xmlns="urn:ieee:std:2030.5:ns">

      (More included XML tags and content)

</DERSettings>
```

The URI of "/edev/1/der/1/derg" is based on the structure presented in the WADL. The example URI relates the resource to a specific end device (number 1) which is a DER (/edev/1/der). An integrated DER system could consist of a solar PV array and a storage unit presented as a single end device but with individual DER resources. The resources for the specific DER must be located (/edev/1/der/1). Then the identifier "derg" locates the specific resource (i.e., DERSettings). These locator names could be used by the server and are used for SEP2 examples, but the entire construct is just a unique URI and the server could use its own convention.

If a HAN device is not a server which is capable of hosting its own resources (a self-device), it must place a copy of each of its resources on a host server. This makes the resources available to that client device and to other authorized client devices on the network. A client device uses the GET method to access resources on a host server and this requires a unique URI, as shown in the previous example to retrieve the DERSettings for a specific DER device. The host server assigns a unique URI to each resource of each client (end-device) that it hosts. Client devices use a resource discovery process to find the URIs on the host server for specific resources of interest.

The process of defining which network server will host the resources of a specific client device is all part of setting up the network and there are many possible system architectures. This is not necessarily a plug and play process, at least for the initial configuration. This is not unlike setting up a home entertainment network. For this discussion, it must be assumed that the client device "knows" which server will host its end-device resources. A SEP2 host server will even setup the URIs for empty resources for a specific end-device. Even though DERSettings for a specific DER may not be placed on the host server for this DER, the server will have an established a URI for the DERSettings of this specific DER. The client DER device will discover the URI where it must place its DERSettings resource.

The HTTP PUT method is used to place a basic resource on the host server and also to update it. An example of a client DER placing its DERSettings resource on the host server is shown below:

```
PUT /edev/1/der/1/derg HTTP/1.1
Host: {hostname}

Content-Type: application/sep+xml

<DERSettings xmlns="urn:ieee:std:2030.5:ns">

      (More included XML tags and content)

</DERSettings>
```

The end device server hosting the DER resources responds with:

```
HTTP/1.1 204 No Content
```

This confirms that the server accepted the placement of the resource.

A list is a special type of resource. It is a container for individual resources. Some resources of a client are basic and only one copy can exist on the server for a specific end device. DERSettings is such a resource. A client device can also create resources, such as a flow reservation request, where several different ones may need to be placed on the host server and more than one maintained. Because there are multiple instances of a resource on the server, these are placed on a list. The POST method is used to place a resource on a list as shown in the example below:

```
POST /edev/1/frq HTTP/1.1
Host: {hostname}

<FlowReservationRequest xmlns="urn:ieee:std:2030.5:ns">

  (More included XML tags and content)

</FlowReservationRequest>
```

The end device server hosting the DER resources responds with:

```
HTTP/1.1 201 Created
Location: /edev/1/frq/3
```

In this case the URI for the specific flow reservation request is returned. This is shown as the third on the list (although the URI would not necessarily follow this convention; it only needs to be a unique URI). If the posted flow reservation request needed to be updated, an HTTP PUT would be used to the URI of the specific resource—not the URI of the list.

4.4.4 SEP2 Class Diagrams

The SEP2 model is based on Unified Modeling Language (UML) principles of classes and objects and their attributes. There are many UML class diagrams in the SEP2 standard and these are handy guides to the structure of the model. They are consistent with the guidelines for UML class diagram and therefore only some of the features will be discussed here and not all of the options. These diagrams collectively define all of the information content in the model and show two types of relationships between the various classes in the model. The diagrams show how one class inherits properties from another class. It uses three different methods for showing inheritance in these diagrams to make it more interesting for the layman to try to follow. This concept of inheritance is critical to understanding how XML packages are constructed. The diagrams also show associations between classes for other than inheritance of properties. This also relates to the construction of XML packages, but in a different way.

A portion of the class diagram for DER Control is shown in Figure 4. Each class is defined by a box. The name of the class is placed in the top center of the box. If the class has attributes, these are placed in the lower portion of the box and separated from the name by a line. The focus of this diagram is on the DERControl class. SEP2 shows the class diagrams, but in the text refers to DERControl as an object rather than using the term class, which is actually more precise. For consistency, the term object will be used here. The intention here is to describe the features of the class diagrams and not deal with the purpose of the specific objects and their attributes.

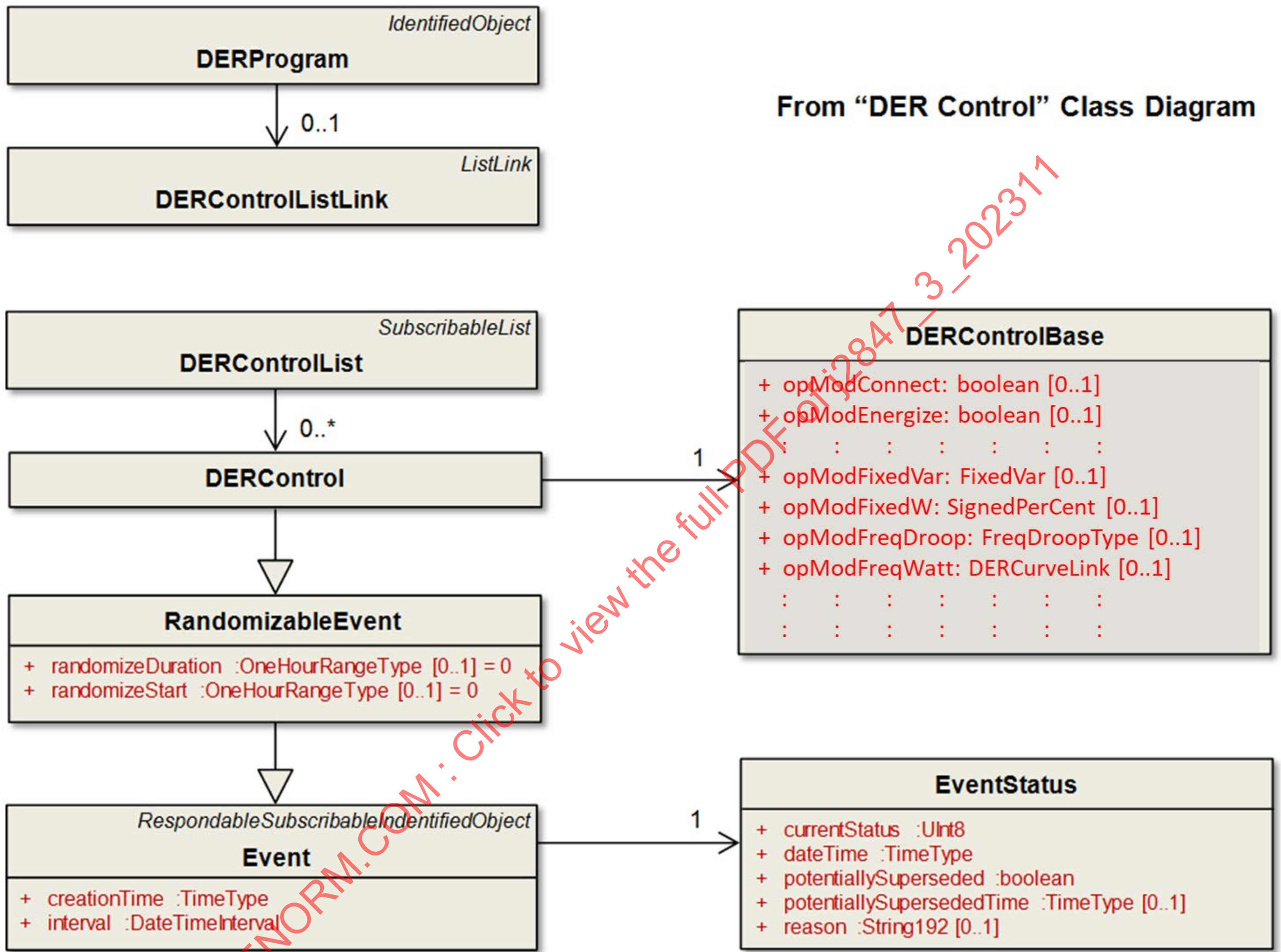


Figure 4 - Sample class diagram for DER control

If a copy of a DERControl object is requested by a client from a server that hosts this resource, the diagram helps show what comes from the server. The DERControl box only shows the name of the object. An arrow with an open head points from the DERControl object to a DERControlBase object. The line with open arrow head is called an association which can be thought of as a "belongs to" or "comes with" relationship. This means that the DERControlBase object and its included attributes are provided with the DERControl object, which itself has no attributes.

There is a "1" next to the arrow and this means that a DERControlBase object must be included, but only one instance. The open arrow with a "0..1" extending from the DERProgram object means exactly zero or one. This effectively means that a DERProgram does not need to have any DERControls, but if it does it can only have one list of DERControls per DERProgram. The open arrow with a "0..*" extending from the DERControlList means zero to whatever (or the list limit defined by SEP2 for this list object). No DERControls are required. If you ask a server for a specific DERControl by its URI you get that specific content. If you ask a server for the DERControlList you get content for from one to many DERControls depending on how the DERControlList resource is requested. There are other relationships that are described in UML books.

The attributes of the DERControlBase object are defined in the lower portion of the box below the object name. Each line describes one attribute. The "+" symbol means it is a "visible" attribute (which is the most common in SEP2) and can be accessed by an outside relating class. There are other symbols which can be found in a UML book. Next the name of the attribute appears. The attribute name may be followed by another name separated by a colon. This name defines the data type or an object that further expands on the data type (which will also be shown later). The "[0..1]" symbol means that the attribute is optional. Otherwise it is mandatory if the object is used. The "randomizeDuration" attribute in the RandomizableEvent Object shows a "=0" at the end. This defines a default value of zero for this attribute if it is not explicitly defined.

The diagram shows a closed arrow extending from the DERControl Object to the RandomizableEvent Object. This means that the DERControl Object inherits all of the attributes from the RandomizableEvent Object. The XML payload for a DERControl resource will include all of the attributes of the RandomizableEvent Object and the Event Object as XML elements. The EventStatus object and its attributes will also be included as elements in the XML document because of the association with Event.

In the upper right corner of the Event Object box, *ResponableSubscribableIdentifiedObject* appears in italics. This is another way of showing inheritance instead of adding boxes and connecting them with closed arrows. This is shown on left side of Figure 5. In a diagram for the Identification Class, the ResponableSubscribableidentifiedObject Object can be found. And it inherits from the RespondableResource Object which in turn inherits from the Resource Object. This means the attributes from all of these objects are also packaged with XML payload for the DERControl resource. Each of these objects shows a symbol "<<XSDattribute>>" which means that any attributes listed below are not object attributes which become XML elements but are actual XML attributes. As discussed earlier the XML attributes appear in the start-tag of the DERControl element.

In the DERControlBase Object there is an attribute named opModFixedW. This is considered to be a SignedPerCent. The SignedPercent is defined in the Types class diagram as shown in the upper right corner of Figure 5. SignedPerCent is defined as an <<XSDsimpleType>> attribute of Int16 which is a defined XML class of "short." The DERControlBase Object also shows an opModFixedVar attribute which uses the FixedVar Object to define it. This object is defined as part of the diagram for DER Control Types and is shown in the lower right corner of Figure 5. Here the single attribute connects with two supporting attributes: one for value which is a SignedPerCent and another for a refType which is of the DERUnitRefType class (an uint8 data type).

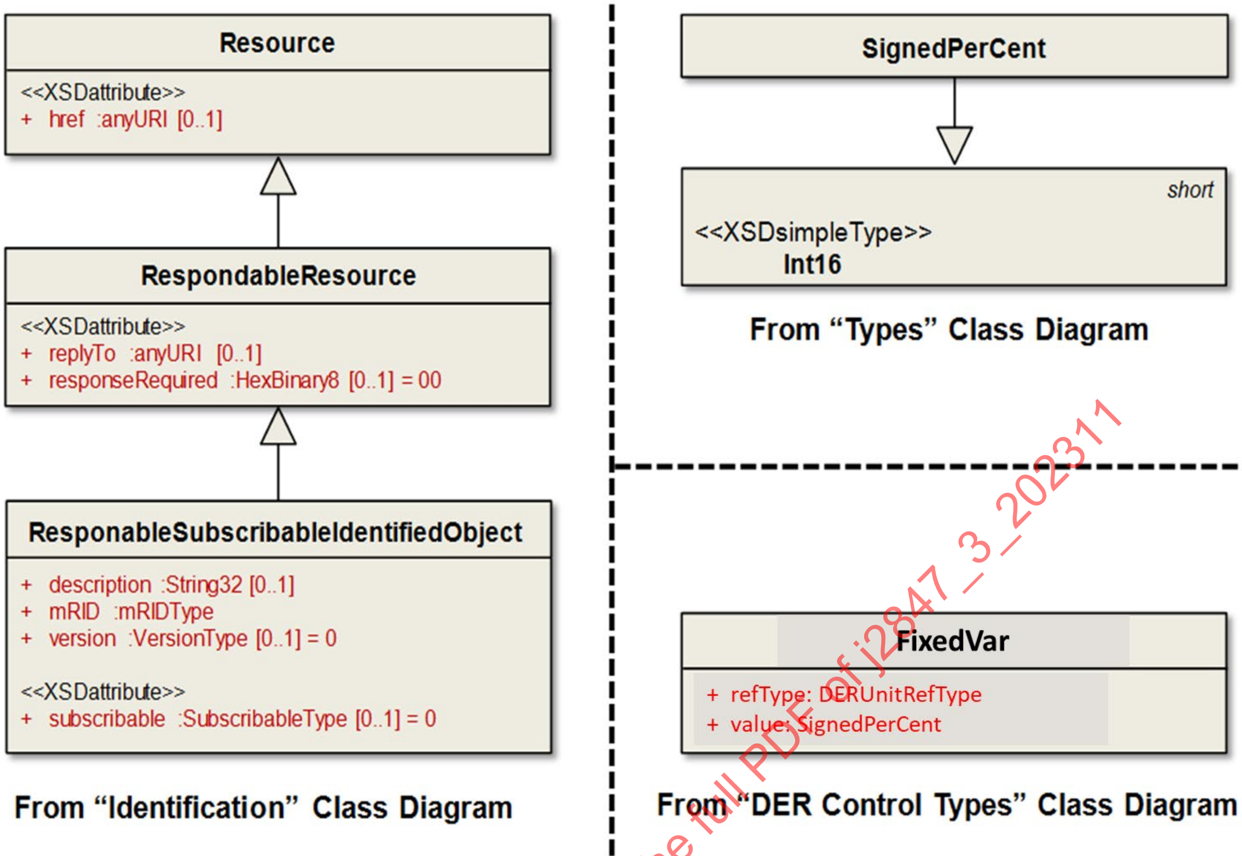


Figure 5 - Class diagrams connected to DER control

The class diagrams have a third method for showing inheritance. An example is shown in Figure 6, which is a portion of the class diagram for flow reservation. The FlowReservationResponse object box actually shows the attribute expansion for the inheritance hierarchy. The double colon ahead of the italicized object name (*::Event*) shows the actual object attributes associated with the object. The XML attributes are shown in the lower portion. This is a compact way to show inheritance, but it is not entirely correct in that it did not bring across the attributes from the associated EventStatus object.

Figure 6 also shows an example of a note which is connected to the RequestStatus object with a dotted line. This defines that the FlowReservationRequest could be an actual request (0) or a cancellation of a request (1) that has already been posted. The model text will define this in detail, but sometimes the class diagram includes some additional information about an object or attribute.

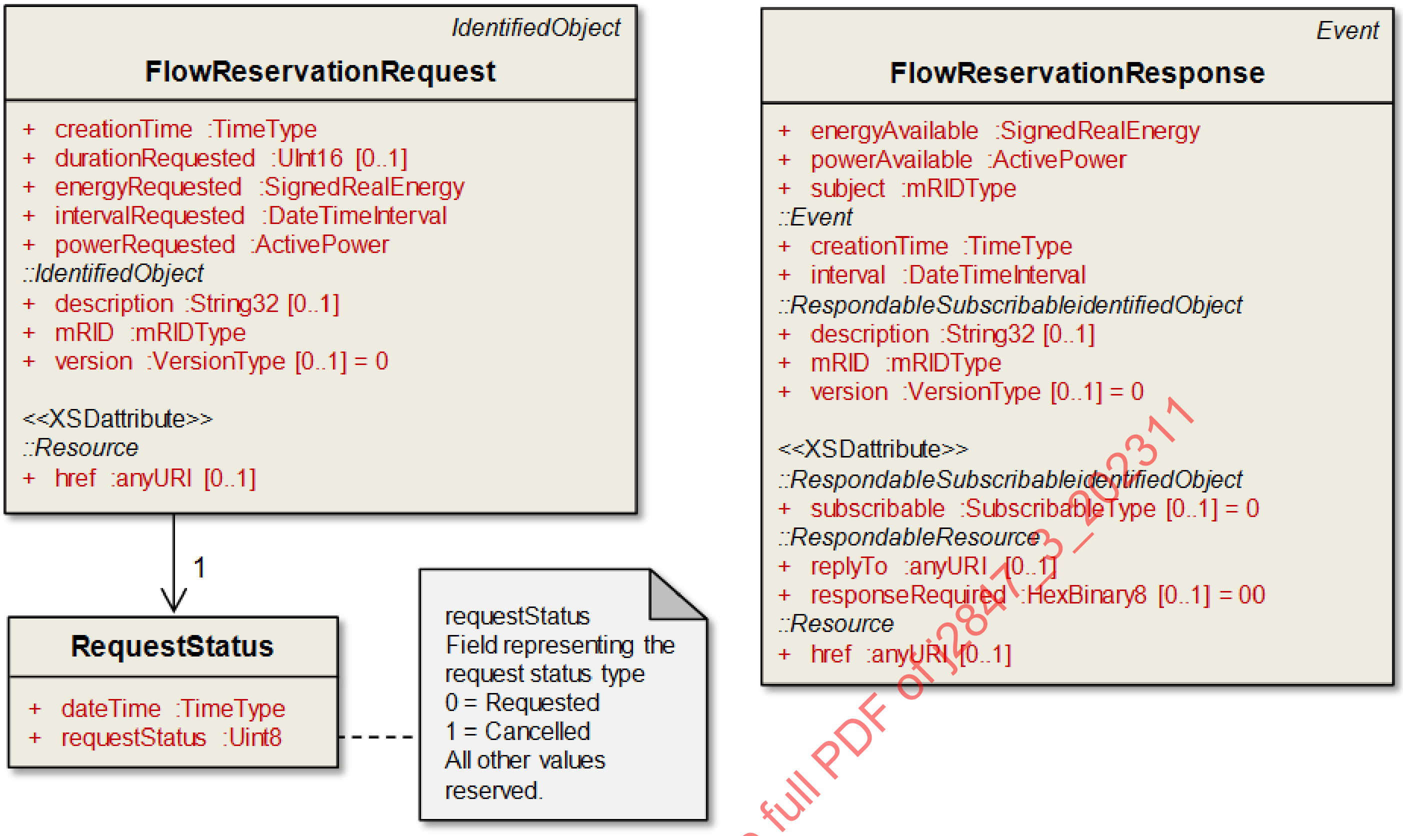


Figure 6 - Excerpts from flow reservation class diagram

4.4.5 SEP2 Informative Model

The SEP2 standard provides an “informative” model in an appendix which can also be found as a separate document. The “normative” (authoritative, real thing) model is constructed using XML and is contained in the SEP2 XSD which is more friendly for computer reading than human reading. There is also an available UML model which was used to create the SEP2 XSD which can be viewed using special software tools. Even though the UML is the source for everything, it is considered to be informative and the XSD is normative. The XSD is actually used as part of the process for encoding and decoding the SEP2 XML messages and therefore needs to be the authoritative source rather than the UML model which created it. This section will provide some guidance to the structure and organization of the text of the informative model. The UML class diagrams provided in this model have already been discussed.

The model begins by defining objects that are widely used by many, if not all, SEP2 function sets. This is where the basic Resource Object, Event Object, List Object, and Link Object are defined. This is also where type objects which are used in more than one function set are defined, such as the PerCent and RealEnergy objects. Every user needs to be familiar with this front portion. The remaining sections of the document are organized by function set which are called “packages” in the model. There is a DER Package, a FlowReservation Package, and a PowerStatus Package which are the primary focus herein. The model ends with a description of the Links Package which also supports all of the function sets. The associated class diagrams are placed near the start of each package.

An excerpt from the informative model is shown below for a portion of the DERControlBase Object which was shown in the class diagram in Figure 4.

DERControlBase Object ()
Distributed Energy Resource (DER) control values

opModFixedW attribute (SignedPerCent) [0..1]
The opModFixedW function specifies a requested charge or discharge mode setpoint, in %setMaxChargeRateW if negative value or %setMaxW or %setMaxDischargeRateW if positive value (in hundredths)

opModFixedPFInjectW attribute (PowerFactorWithExcitation) [0..1]
The opModFixedPFInjectW function specifies a requested fixed Power Factor (PF) setting for when active power is being injected. The actual displacement SHALL be within the limits established by setMinPFOverExcited and setMinPFUnderExcited. If used simultaneously with other reactive power controls (e.g. opModFixedVar) the control resulting in least var magnitude SHOULD take precedence.

This excerpt only shows the opModFixedW and opModFixedPFInjectW attributes. The only new information beyond what is shown by the class diagram is the definition of the object and the attributes.

The RespondableResource Object, which is shown in the class diagram in Figure 5, is shown below:

RespondableResource Object (Resource)
A Resource to which a Response can be requested

replyTo attribute (anyURI) [0..1] «XSDattribute»
A reference to the response resource address (URI). Required on a response to a GET if responseRequired is "true."

responseRequired attribute (HexBinary8) [0..1] «XSDattribute»
Indicates whether or not a response is required upon receipt, creation or update of this resource. Responses shall be posted to the collection specified in "replyTo."

This resource only has XSDattributes. It also shows that in inherits from the Resource Object which is shown in paranthesis in the first line.

4.4.6 Web-Application Description Language

The SEP2 Web-Application Description Language (WADL) defines the recommended URI structures to be used to interact with specific resources in the model. Not all SEP2 resources can be directly accessed and the WADL defines which resources can be accessed. The WADL is organized by function set.

An example from the WADL is shown below for the DERSettings resource:

DERSettings Resource

The DER settings of the associated EndDevice or SelfDevice

Sample URI: /edev/{id1}/der/{id2}/derg
Request Representation: DERSettings
Response Representation: DERSettings
Methods: GET/HEAD: Mandatory, PUT: Mandatory, POST: Error, DELETE: Error

The first line is the title of the resource as used in the model (e.g., DERSettings). This is followed by a brief description, which in this case notes that this resource contains the DER settings for the associated end device or self device.

A sample URI is provided in the next line. The WADL uses descriptive abbreviations to define the unique URI and these could be used by a server, but a server could have its own convention. This resource belongs to an end device (edev). The host server could contain resources for many end devices on the HAN. The symbol {id1} signifies that some identification is needed for each end device (which could just be as simple as a number 1, 2, 3, etc.). This resource is related to a specific DER of an end device which may have more than one DER integrated into it. The "/der/{id2}" is used to designate a specific DER. Examples in this document will use "der/1" because a PEV is a single DER. The "/derg" identifies the specific DERSettings resource. Again, this makes it easier to follow the model, but the server could just create a unique URI for this resource for a specific DER entity for a specific end device.

When a GET is used with the URI, the Response Representation designates which resource will be returned with the XML payload. The Request Representation shows which object is associated with the PUT, POST, and DELETE methods. In this example, both are the same. This is typical for a basic resource. When list resources are used these will generally be different. The request will be the named object and the response will be a list object that acts as a container for the named object.

The last line shows the compliance (MODE) definitions for each of the HTTP methods: Mandatory ("M") must implement and conform; Optional ("O") may implement, and if implemented, must conform; Discouraged ("D") should not implement, but if implemented, must conform; and Error ("E") must return one of the specified response status codes (400 - Bad request or 405 - Method Not Allowed).

A summary of WADL entries for resources associated with the DER, flow reservation, and power status function sets is provided in Table 2. The table is organized in the order of function set. Within the DER function, DER Program resources (derp) are placed ahead of end device resources (edev). The WADL resources shown in red are not actual resources in the SEP2 model. These act like a specialized search of a more basic resource, such as the DERControlList response for the ActiveDERControlList resource. The provided list would only be for those DER Controls with an active status. The resources with a sample URI of "edev" are associated with a specific HAN device. The resources with a "derp" URI do not belong to the end device. These belong to the DER controller and are placed and maintained on its server using embedded software and not SEP2 messages.

Table 2 - WADL summary for selected resources

Resource	Sample URI	HTTP Method				GET Response
		GET	PUT	POST	DEL	
ActiveDERControlList	/derp/{id1}/actderc	M	E	E	E	DERControlList
DefaultDERControl	/derp/{id1}/dderc	M	O	E	E	DefaultDERControl
DERControl	/derp/{id1}/derc/{id2}	M	E	E	O	DERControl
DERControlList	/derp/{id1}/derc	M	E	O	E	DERControlList
DERCurve	/derp/{id1}/dc/{id2}	M	E	E	O	DERCurve
DERCurveList	/derp/{id1}/dc	M	E	O	E	DERCurveList
DERProgram	/derp/{id1}	M	E	E	O	DERProgram
DERProgramList	/derp	M	E	O	E	DERProgramList
AssociatedDERProgramList	/edev/{id1}/der/{id2}/derp	M	E	O	E	DERProgramList
CurrentDERProgram	/edev/{id1}/der/{id2}/cdp	M	E	E	O	DERProgram
DER	/edev/{id1}/der/{id2}	M	O	E	O	DER
DERAvailability	/edev/{id1}/der/{id2}/dera	M	M	E	E	DERAvailability
DERCapability	/edev/{id1}/der/{id2}/dercap	M	M	E	E	DERCapability
DERList	/edev/{id1}/der	M	E	O	E	DERList
DERSettings	/edev/{id1}/der/{id2}/derg	M	M	E	E	DERSettings
DERStatus	/edev/{id1}/der/{id2}/ders	M	M	E	E	DERStatus
FlowReservationRequest	/edev/{id1}/frq/{id2}	M	M	E	O	FlowReservationRequest
FlowReservationRequestList	/edev/{id1}/frq	M	E	M	E	FlowReservationRequestList
FlowReservationResponse	/edev/{id1}/frp/{id2}	M	M	E	D	FlowReservationResponse
FlowReservationResponseList	/edev/{id1}/frp	M	E	D	E	FlowReservationResponseList
PowerStatus	/edev/{id1}/ps	M	M	E	E	PowerStatus

4.4.7 Resources, Links, and Lists

Figure 7 shows a small portion of the inheritance hierarchy for objects in SEP2. The left side shows resources and the right side shows links. The focus is on those objects which are of interest for flow reservation and DER applications. The objects shown in red are associated with the DER function, those shown in blue are associated with the flow reservation function, and those in green are associated with power status. Those objects shown in bold will be discussed in much more detail later in this document.

There are two fundamental types of objects in the model: Resource objects and Link objects. A resource is an addressable unit of information—actual content. Links only provide a reference, via URI, to another resource.

Resource

- **DERCapability (edev)**
- FunctionSetAssigmentsBase
    - FunctionSetAssigments (edev)
- IdentifiedObject
    - DERCurve (derp)
    - DERProgram (derp)
    - **FlowReservationRequest (edev)**
- List
    - DERCurveList (derp)
    - DERList (edev)
    - DERProgramList (derp)
    - FlowReservationRequestList (edev)
- **PowerStatus (edev)**
- RespondableResource
    - ResponableSubscribableIdentifiedObject
        - Event
            - **FlowReservationResponse (edev)**
            - RandomizableEvent
                - **DERControl (derp)**
- SubscribableIdentifiedObject
    - DefaultDERControl (derp)
- SubscribableResource
    - AbstractDevice
        - EndDevice (edev)
        - SelfDevice (sdev)
    - DER (edev)
    - **DERAvailability (edev)**
    - **DERSettings (edev)**
    - **DERStatus (edev)**
    - SubscribableList
        - DERControlList (derp)
        - EndDeviceList (edev)
        - FlowReservationResponseList (edev)
        - FunctionSetAssignmentsList (edev)

Link

- CurrentDERProgramLink
- DefaultDERControlLink
- DERAvailabilityLink
- DERCapabiltyLink
- DERSettingsLink
- DERStatusLink
- ListLink
    - ActiveDERControlListLink
    - AssociatedDERProgramListLink
    - DERControlListLink
    - DERCurveListLink
    - DERListLink
    - DERProgramListLink
    - FlowReservationRequestListLink
    - FlowReservationResponsetListLink
- PowerStatusLink
- TimeLink

Figure 7 - Object hierarchy

After the name of each resource object is a designation of whether it is an end device (edev) resource or a DER program (derp) resource. The end device resources belong to a specific device. The DER program resources belong to the DER controller and could be retrieved and used by many devices. It could be that a DER program only applies to a single DER, but a program could be set up to apply to many. The grouping is flexible because a DER controller may want to organize DER devices by groups and provide a common control signal to each group.

A list object can be thought of as a container to hold a collection of resource object instances. An example is shown in Figure 8 for the DERControlList resource. The example shows four individual DERControl instances which all belong to the DERControlList, but only in the sense of a relational database. These are the only DERControl objects in the server. Each of the four DERControl objects has a unique URI. Rather than use a simple numerical sequence for successive DERControl URIs, which is dangerous to assume, this example uses a code for frequency-watt curve, volt-var curve, and fixed-flow commands. Otherwise one might think that if the active control is URI/2 that the next one could be URI/3 and this could be wrong. Each list object has a defined sort order of primary, secondary, and tertiary attributes of the basic object. These are all defined in the SEP2 standard. Each DERControl object is an Event which has a designated start time, which is expressed as the integer number of seconds since midnight on January 1, 1970, GMT (class TimeType). For simplicity in this example only the last three digits are shown. For the DERControlList, the DERControl objects are sorted by ascending order event start time, and then descending order of creation time, and then by descending order of mRID.

List Query

```
GET /derp/1/derc?s=x&a=y&l=z HTTP/1.1

<DERControlList … all="4" result="value" … >
```

	GET	result=	XML Payload(s)
1	URI	1	A
2	URI?	1	A
3	URI?s=0&l=1	1	A
4	URI?s=1	1	B
5	URI?l=2	2	A,B
6	URI?s=2&l=2	2	C,D
7	URI?a=xx200	1	C
8	URI?a=xx200&l=2	2	C,D
9	URI?s=1&a=200&l=2	1	D

Direct Access

```
GET /derp/1/derc/ff2163

<DERControl … >
```

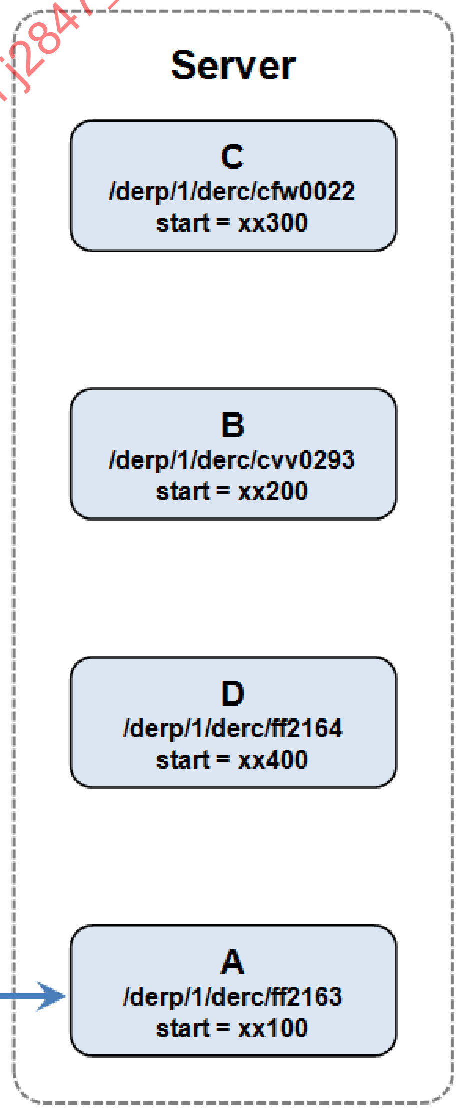


Figure 8 - List query example

There are two ways to request a DERControl. One is to directly request the exact DER Control using its full URI, if it is known by the client. In this example DERControl "A" could be retrieved from the server using:

```
GET /derp/1/derc/ff2163 HTTP/1.1
Host: {hostname}
```

The server would respond with the XML payload for this exact DERControl.

The other method is to retrieve the content of the DERControlList by using a GET query of the list. The XML payload would return content for potentially several DERControl instances. It is more likely that the exact URI of a specific DERControl would not be known and a list query would be used to retrieve the one(s) of interest from the server. An example of a list query is shown below:

```
GET /derp/1/derc?s=0&a=xx200&l=2 HTTP/1.1
Host: {hostname}
```

The server would respond with the XML payload for the DERControlList which would include, in this case, the XML payload for DERControls "C" and "D."

```
HTTP/1.1 200 OK
Content-Type: application/sep+xml
Content-Length:{contentLength}

<DERControlList all="4" results="2" href="/derp/0/derc" subscribable="1"
        xmlns="urn:ieee:std:2030.5:ns" >
  <DERControl>
       (XML content for DERControl C)
  </DERControl>
  <DERControl>
       (XML content for DERControl D)
  </DERControl>
</DERControlList>
```

The query string parameters (s,a,l) are described in detail in the SEP2 standard. The DERControlList figure shows several examples. Query string parameters are parameters added to a URI to provide filtering of list items returned in query results. The "s" (start) indicates the first ordinal position in the list to be returned in the query result list as determined by the list's ordering with zero being first. Zero is also the default if this parameter is not provided. The "l" (for limit, not one) is used to set the maximum number of list items (up to 255) to be included in the query result list. If this query string parameter is not specified, the default limit is one. The "a" (after this time) is used to indicate that only items whose primary key occurs after the given date time parameter should be included in the query result list. This query string parameter is only applied to list resources that are ordered using a time-based primary key. If both a "start" and "after" query string parameter are used simultaneously, the "after" query string parameter has precedence, and the "start" position zero is relative to the position specified by the "after" parameter. Null queries are possible.

One important question is how a client device can first locate the URI of the DERControlList on the server of a DER controller. A related question is how the DER controller (as a client) can find the DERSettings resource on the end device server for the PEV. This is all done by a discovery process during which the respective client device first locates the URI of the DeviceCapability resource on the server. SEP2 makes this the "master" discoverable resource. Once the URI of this resource is located, SEP2 provides a structure of link resources and listlink resources to map the URI structure of the end device or DER program. An example of this is shown in Figure 9 for the DERSettings (edev) resource and the DERControl (derp) resource.

The WADL shows a meaningful sample URI structure for all resources. DERSettings, for example, has a sample URI of "/edev/1/der/1/derg." However, the actual URI could be "/d43x97yz3"—not great, but SEP2 requires that all resources should be capable of being found without relying on a URI naming convention. The WADL samples are great and a SEP2 server could follow the sample structure, but a client should not rely on this convention. A client device should discover and use the links, listlinks, list resources, and pure resources to identify the actual URI structure for use in its GET, PUT, and POST methods. This way, a GET /d43x97yz3 will return the DERSettings XML payload and it is exactly equivalent to the sample GET /edev/1/der/1/derg request, but with the actual URI for this resource.

The class diagrams in the SEP2 model show all of these connections. Figure 9 just shows the path to DERControl and to DERSettings. It may take multiple GET requests to pure resources to find links or listlinks. Lists will need to be searched, such as the DERList to find a specific DER. In some cases, the inheritance of objects will bring information into the XML payload of objects—such as the AbstractDevice XML being returned to a client from a GET of a specific EndDevice object URI. When a PEV connects to the host server and application server there is some housekeeping work to be done to locate all of the resources needed for a session. This process is beyond the scope of this document.

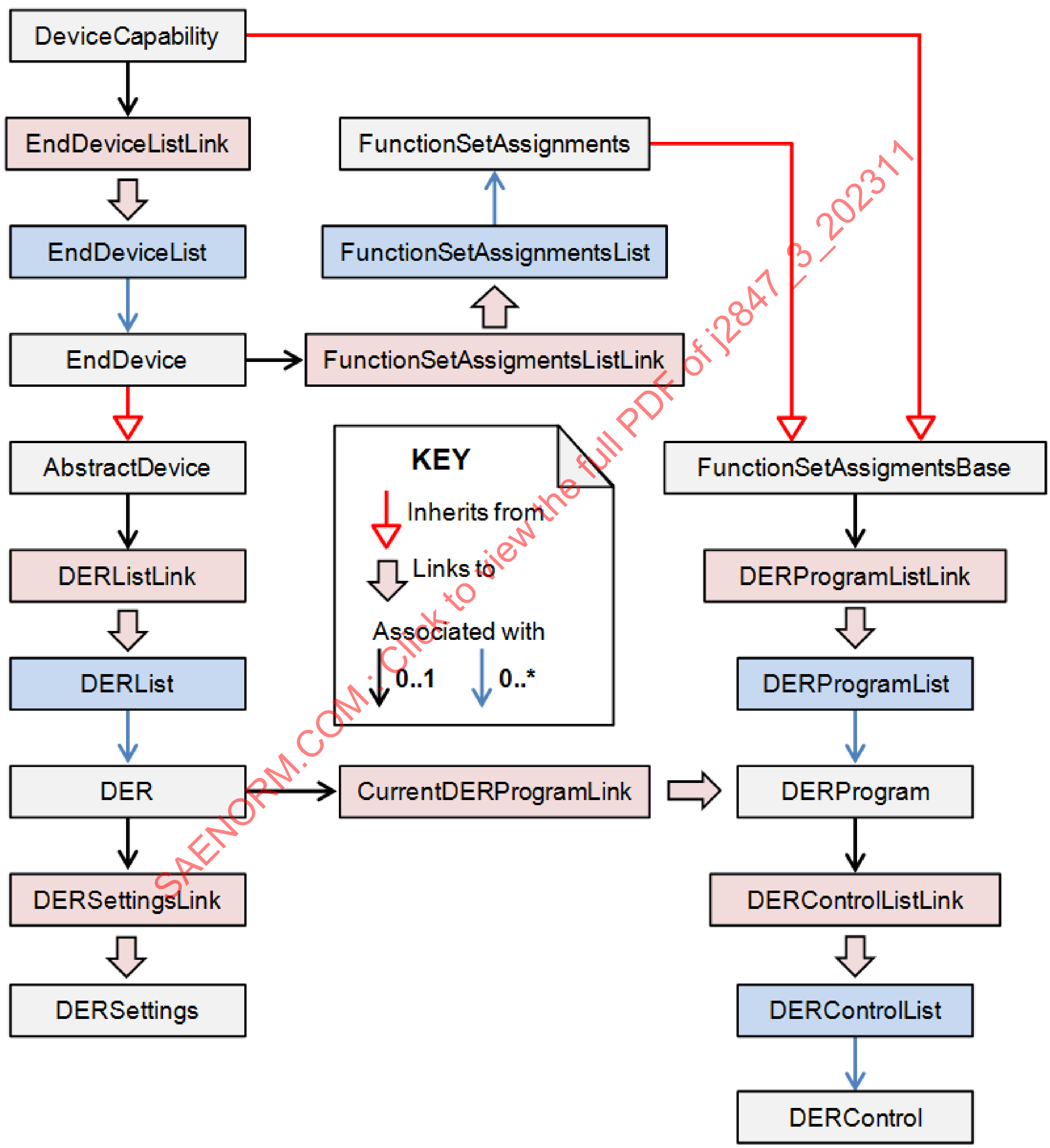


Figure 9 - Connecting resources

4.5 EVSE and PEV Network Options

Smart Energy Profile 1.0 was created to deal with home energy networks consisting of primarily simple devices such as thermostats and on/off power switches. These are low cost client devices. The object model has now been extended in SEP2 to include the information requirements for complex devices, such as a storage DER system. A stationary DER system would most likely host its own resources as a self-device. However, it is more likely that a mobile DER (PEV) would only be a client device and maintain its resources on an associated end-device server.

Most client devices are part of a fixed network. The decisions for hosting client device resources can be made when the HAN is first setup and when the homeowner enrolls in different applications. This is sometimes called an “out of band” process and it is not generally an automatic “plug and play” action. It may involve data entry on the device and in associated servers either in the home or at the service provider. It could use other communication protocols than SEP2. For a PEV at home, this would be no more difficult than setting up any other fixed complex device, such as a solar PV system. This gets much more complex if the PEV is not at home. Reverse power flow is also subject to utility approval at the point of connection and this may preclude any simple drive up and discharge operation even if a client PEV had the capability to directly communicate with an aggregator and host its resources on the aggregator server.

No matter how it is accomplished between an EVSE and a PEV, a PEV with an onboard inverter must be able to access SEP2 XML messages for DER Controls. The EVSE could serve as a simple bridge to the HAN, in which case the PEV could retrieve DER Controls from a compatible DER controller on the HAN or beyond. The EVSE could also act as a SEP2 gateway for the PEV and be the server for the PEV end device resources and the host for the DER Program of the DER controller. There are many possible SEP2 networks, but only two will be considered for examples in this document: one based on an EVSE bridge and another based on an EVSE gateway.

The system shown in Figure 10 uses an EVSE that provides a bridge for the PEV to directly connect to the HAN. In this example, it translates the PLC messages to a HAN protocol, such as Ethernet, but the EVSE does not interact with the message payload. Both the EVSE and the PEV are client devices and therefore must host their resources on a discoverable server. The Energy Services Interface (ESI) performs this function for this network and acts as the end device server for the PEV and EVSE. The PEV must place copies of its resources on the ESI server using the HTTP PUT method. It is assumed that the PEV can directly discover the resources on the EMS and DER Controls (DERC) can be retrieved by the PEV using the GET Method or by subscription.

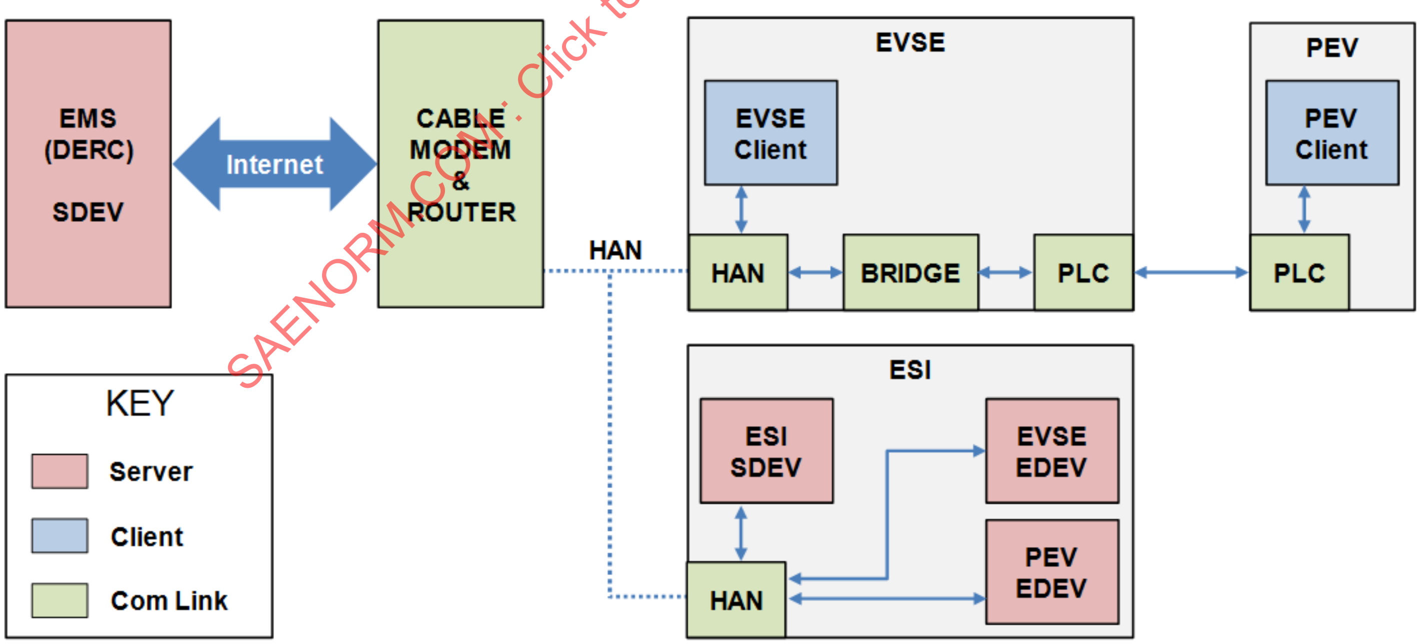


Figure 10 - EVSE as bridge

The system shown in Figure 11 does not provide direct access to the HAN by the PEV. The PEV can only communicate directly with the EVSE using the SEP2 protocol. This is a private point-to-point network. In this system, the EVSE acts as an end-device server for the client PEV to host its resources. The PEV would discover the EVSE server and engage with it just as it would have engaged with the ESI in the bridge system. The DER controller will be based on the IEC 61850 DER Model, but it may or may not use SEP2. If the EVSE engages with the DER controller using the DER controller protocol, the EVSE will act as a SEP2 gateway between the DER controller and the PEV. If the DER controller and EVSE both use the SEP2 protocol, the gateway functionality is much less complex. This is all beyond the scope of this document.

If the gateway is well designed, the PEV would appear to be connected using the virtual architecture shown in the lower right of the figure. This would look like a point-to-point exchange between the DER controller and the PEV using SEP2. The gateway must ensure that resources placed in the PEV end-device representation are immediately made available to the DER controller. It must also ensure that DER Controls from the DER controller are retrieved and presented to the PEV with minimal transport delay and at the update rate of the DER application.

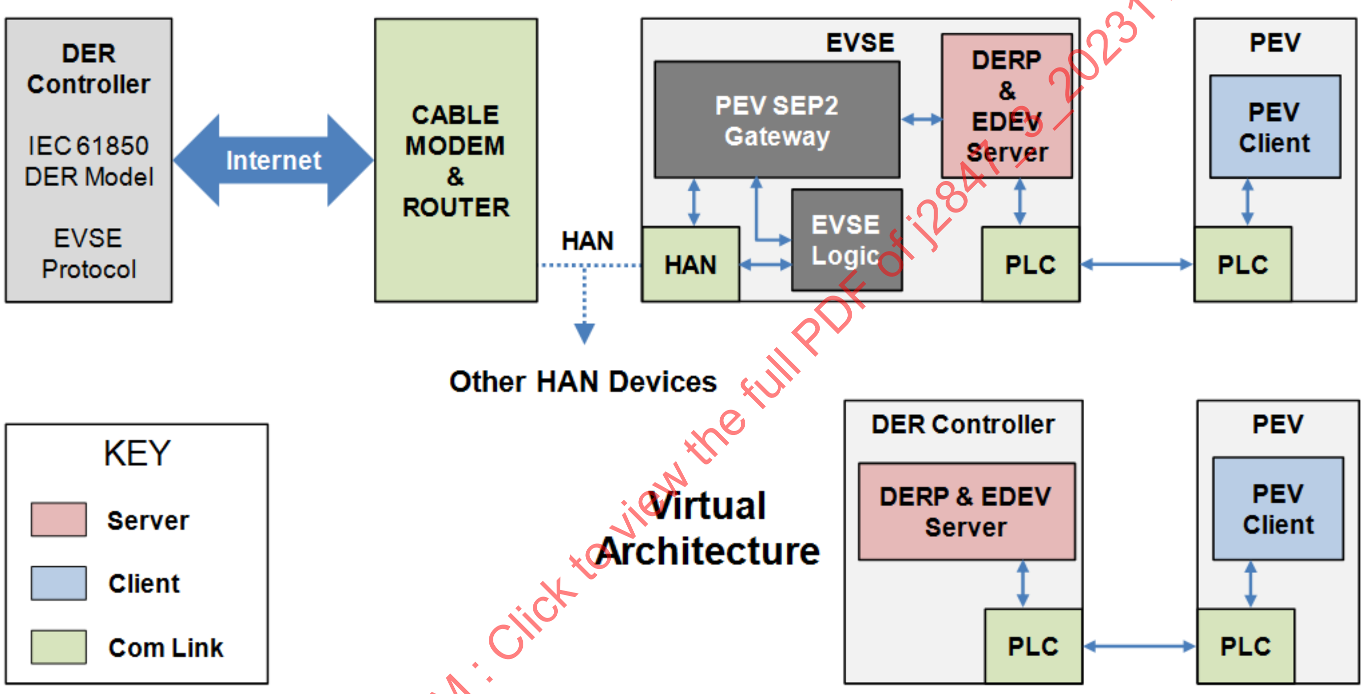


Figure 11 - EVSE as gateway

One key factor in setting up the network is getting approval to interconnect a DER device. A homeowner that wishes to connect a DER device in parallel with the grid must apply to the distribution utility for authorization, follow the utility procedures to secure their approval, and enter into an interconnection agreement with the utility. The application will be for a specific inverter (or PEV) model that has been tested and listed by a listing organization that is acceptable to the local utility to a standard (such as UL 1741) that is also acceptable to the local utility. There is currently no process for third-party-authenticated electronic verification that the inverter model is listed and approved by the governing interconnect agreement for the EVSE. Until such a process exists, it will be prudent for the EVSE owner to ensure that any vehicles that plan to discharge do so in compliance with their utility interconnection agreement. One way to achieve this is to provide controls and displays onboard the vehicle to authorize discharging at that EVSE.

Discharging at work or at public locations becomes much more complex for PEVs with onboard inverters. The interconnection procedures all assume fixed DER devices, not mobile devices. Even if individual model PEVs are all listed, there is no established method to approve a random mix of listed inverter models at a site. Significant work will need to be done with state regulators to establish procedures to allow for public sites to allow interconnection of a random mix of listed PEVs. It may be required that each PEV provide its model number to the site ESI or EMS to check against a list of approved models. SEP2 provides an object that can be used for this purpose.

4.6 Harmonization of DER Information Models

The SEP2 information object model is described by an XSD that contains the definitions of the resources, attributes, and elements as well as their textual descriptions. This XML schema is to be used for encoding and decoding XML messages to and from the PEV. An informative textual extract of the XSD is provided as an appendix in the Application Protocol Standard.

The international standard IEC 61850-7-420 describes the power control functions for inverter-based DER systems, including photovoltaic (PV) systems, battery storage systems, PEVs, and any other DER systems with controllable inverters. It defines the IEC 61850 information models to be used in the exchange of information between these inverter based DER systems and the utilities or other entities which are tasked with managing the active and reactive power capabilities of these inverter-based systems.

SAE J2836/3 incorporated most, but not all, of the inverter functions defined in IEC 61850-7-420. It included only those functions that were also included in the SEP2 DER function model. The SEP2 inverter model was based on the SunSpec Alliance Inverter Control Model which was also derived from IEC 61850-7-420. Because the IEC, SAE, SunSpec Alliance, and SEP2 information models were under development at the same time, some differences exist between the information models.

SAE J2836/3 defines information requirements for two utility use cases: U6 (Basic Distributed Energy Resource) and U7 (Advanced Distributed Energy Resource). Table 3 shows these use cases. Utility use case U5 (Optimized Energy Transfer), which is defined by SAE J2836/1, is also shown. The use case and associated inverter modes (functions) are listed in the first column. U5 and U6 are single functions. U7 consists of several modes. The second column lists the type of function: fixed command, autonomous curve, or specification curve function. The third column (SEP2) designates the SEP2 function set that supports the use case mode: either DER or flow reservation (FR). The VAr column lists whether the inverter must be capable of sourcing or absorbing reactive power to perform the function.

Table 3 - SAE use cases and modes

Use Case and Mode	Type	SEP2	Var
U5 Flow Reservation	Fixed	FR	
U6 Fixed W	Fixed	DER	
U7 Fixed Power Factor	Fixed	DER	Y
U7 Fixed Var	Fixed	DER	Y
U7 Frequency-Watt	Auto	DER	
U7 Volt-Watt	Auto	DER	
U7 Volt-Var	Auto	DER	Y
U7 Watt-Power Factor	Auto	DER	Y
U7 Low Voltage Ride Through	Spec	DER	
U7 High Voltage Ride Through	Spec	DER	

There is no requirement for a PEV to have an onboard inverter. And even if it does have an inverter, there is no requirement to implement any or all of the inverter functions and associated communication to allow external control. The PEV inverter could be manually engaged to provide basic reverse flow into a live grid. However, if a PEV does implement communication to serve as a DER device, it must at least implement U6. This provides the basic capability to directly control the rate of charging or discharging of the PEV. Whether any of the U7 functions are implemented in a PEV inverter will depend on the economic case for the PEV manufacturer and the PEV customer. Some functions represent new capabilities that have no established market today.

IEC 61850-7-420 was initiated primarily to allow solar PV systems and stationary battery storage systems to assist the distribution utility with managing power quality and loads on feeders as the penetration of renewable sources increase. The autonomous volt-var capability is a key capability for providing voltage support. Unlike PEVs, solar PV systems and stationary battery storage systems are capable of sustained energy production and they have generally much larger power output than a PEV. The low and high voltage ride through functions may be essential for keeping PV solar and stationary storage systems on-line during voltage disturbances to allow the inverter to help correct the voltages using autonomous volt-var curves. While utilities have great interest in this capability for solar PV systems, they may prefer to have a PEV that is sourcing power to just drop offline during a voltage disturbance. There may not be a viable economic model to justify some of the U7 DER functions at the current time for either the PEV or the utility, but this will probably change in the future. SEP2 must provide for solar PV inverters and therefore includes most of the IEC functions. SAE J2836/3 U7 also allowed for the advanced functions, but the PEV manufacturer will need to decide which functions to implement in the inverter.

SAE J2836/3 includes information that precisely defines the PEV charging requirements which can be used by a DER controller to both use the PEV as a DER and also ensure that the PEV achieves its charging objectives. It is not required that a controller always uses this information, but there are applications where this cooperation is desired. Some of this has been added to the SEP2 model but needs to be incorporated in IEC 61850. Others are in both the IEC and SEP2 models but the use by a PEV may be different. This information is shown in Table 4. They are explained in Appendix E of SAE J2836/3 and in the SEP2 model. The SEP2 items are attribute names of the objects listed in parentheses.

Table 4 - Required PEV state information

SAE J2836/3 Information	SEP2 Model	IEC 61850
PEV Charging Requirements		
Time Charge Is Needed	timeChargeIsNeeded (PowerStatus)	**Yes, Needed**
Energy Request	energyRequestNow (PowerStatus)	**Y Needed**
Minimum Charging Duration	minimumChargingDuration (PowerStatus)	**Y Needed**
Maximum Forward Power	maxForwardPower (PowerStatus) - ALTERNATE	Y
Time of Reference	timeChargingStatusPEV (PowerStatus)	Y
Maximum Ratings of DER		
Maximum Forward Power	setMaxChargeRate (DERSettings)	Y
Maximum Reverse Power	setMaxDischargeRate (DERSettings)	Y
Minimum Power Factor	setMinPF (DERSettings)	Y
Maximum Reactive Power	setMaxVar (DERSettings)	Y
Availability of DER Capability		
Duration Maximum Forward Power	maxChargeDuration (DERAvailability)	**Y Needed**
Duration Maximum Reverse Power	availabilityDuration (DERAvailability)	**Y Needed**
Available Reactive Power	statVarAvail (DERAvailability)	Y

To ensure interoperability of the PEV serving as a DER device with a utility DER management system, a customer energy management systems (EMS), or another entity's DER controller, it is essential that the SEP2 model align as closely as practicable with the IEC 61850-7-420 model. While the content of the IEC and SEP2 model may be similar, the model structure may be different. While a DER controller may not use the SEP2 protocol, it will most likely use a protocol that follows the IEC object model. Some device between the DER controller and the PEV will need to serve as a bidirectional protocol translator and this task becomes much easier if the IEC and SEP2 models are aligned.

4.7 Flow Reservation and Reverse Power Flow

There are three methods for engaging a PEV in reverse power flow (discharging): manual control, the SEP2 DER function, and the SEP2 flow reservation function. The primary purpose of this document is to address the use of the DER function for controlling the inverter. However, this section will briefly discuss the use of the flow reservation function for discharging the PEV battery. The primary purpose for the flow reservation function is to manage the charging of the PEV and it was derived from the SAE J2836/1 use case U5, Optimized Energy Transfer. Flow reservation and Optimized Energy Transfer are discussed in more detail in SAE J2847/1.

Manual control of discharging is fully described in SAE J2836/3 and will be briefly described here. When the PEV is connected to a live grid through the EVSE, the vehicle operator may wish to just discharge the battery into the grid. This could be for the purpose of draining the battery before a maintenance action. It could be that a homeowner wishes to discharge a PEV for 2 hours at a specific power level to offset home loads during a peak time. SEP2 communications are not required to perform this discharge. The PEV should provide manual controls and displays to select the start time for the discharge, the duration of the discharge, and the power level for the discharge. The operator should also be able to specify a minimum state of charge and the PEV would stop discharging at the earlier of reaching the specified duration (end time) or the minimum SOC. Under most circumstances there would be no need for the PEV to secure permission from any entity to make the discharge.

The flow reservation function can be setup to operate like the manual control mode, except the scheduled discharge is subject to approval and modification by an authorized approval entity. The PEV must be enrolled with an entity that is authorized to receive a request to discharge from the PEV and respond to the request. This could simply be a home energy management system. The controls and displays provided by the vehicle to setup this mode could be similar to those of manual control. The operator could define a start time, duration (or end time), minimum SOC, and discharge power. Unlike manual control, where the discharge would just be scheduled, under flow reservation the request must be vetted by the authorized approval entity. The approval entity responds to each request and could authorize a lower rate of discharge, a lower amount of energy transfer, a different start time and different duration. Some of the elements of information associated with flow reservation request and flow reservation response are shown in Table 5. The energy requested must be a negative value to indicate discharging. The power requested should always a positive magnitude for both charging and discharging.

Table 5 - Flow reservation function

Flow Reservation Request	**Flow Reservation Response**
Identified Object	**Event Object** interval.start (TimeType) interval.duration (UInt32) currentStatus (UInt8)
FlowReservationRequest Object energyRequested (SignedRealEnergy) powerRequested (ActivePower) durationRequested (UInt16) intervalRequested.start (TimeType) intervalRequested.duration (UInt32) requestStatus (UInt8)	**FlowReservationResponse Object** energyAvailable (RealEnergy) powerAvailable (ActivePower)

Figure 12 shows how some of the elements of information in the flow reservation function are defined for a charging request. The energyRequested is the area under the unconstrained charging profile of power versus time. The energyAvailable area equals the energyRequested, but the powerAvailable is lower than the request and the start time is earlier than requested.

For a charging flow reservation request, the energy, power, and duration requested are calculated by the PEV software based on data entry by the driver of the Target State of Charge and the Time Charge Is Needed. The driver could enter a requested time to start charging but this could also be automatically determined by the PEV software based on time of use tables. The PEV software sets requested duration equal to the time interval between the requested start time and the Time Charge Is Needed.

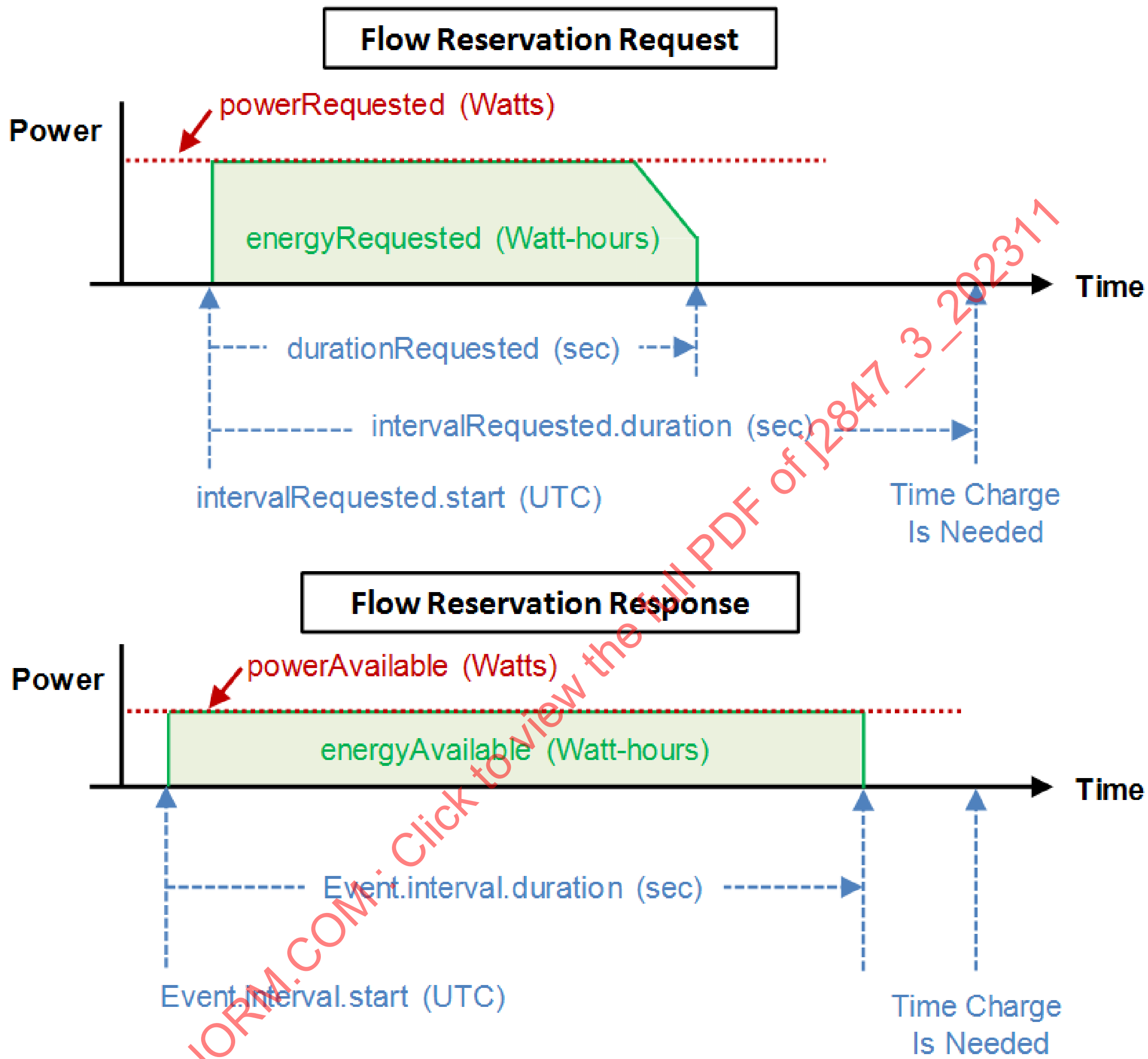


Figure 12 - Flow reservation parameters

For a discharge flow reservation request, the PEV could provide for direct entry of the start time, power level, and duration and the energy request would be calculated. Alternatively, the PEV could accept the start time, power level, and energy request, and the duration request would then be calculated. The requested duration in the flow reservation request is based on the earlier of the time that recharging must start or the time of disconnect (Time Charge Is Needed). This requested duration is used as a planning parameter by the approval entity.

The flow reservation response authorizes the start time and the available power. The response also provides the available energy and the event duration. A PEV can only directly control the start and stop of charging or discharging and the power level. The energy flow is a result of the power flow over time. For charging, the PEV would normally stop charging when it achieves the target SOC and not when the available energy has been transferred or the event duration has been reached. For discharging, the exit condition should be based on the earlier of achieving the available energy transfer, the event duration, or a minimum reserve SOC.

The flow reservation response should never provide a power available that is greater than the power requested and if a higher value is provided the PEV should use the requested value. Also, the PEV should not allow a value for the power requested to be greater than the capacity of the inverter as limited by the EVSE control pilot. If during a transfer the inverter limit is reduced by changes in the control pilot, the lower value of the power available and the limit should be used by the inverter.

The flow reservation function is often presented as a single matched request and response, although it can actually be more flexible than that as will be discussed later. However, for many intended applications a one to one match of request and response may be appropriate. For example, some utilities have expressed concern that a specific transformer could be subject to overload if a cluster of PEVs all charged at maximum rate at the same time. flow reservation could be used to spread out the charging. All of the PEVs could request the start of charging based on a time of use schedule, but the utility could stagger the start times using Flow reservation and reduce the power levels to manage the load. A one time schedule for the evening could be used with each PEV.

It is also possible to enter a sequence of requests. These could be staged by PEV software, where a new request is made when the current request has been completed. Optionally, SEP2 allows up to three requests to be entered into the reservation server. A homeowner could schedule a discharge to start at 5:00 p.m. for 2 hours and also schedule charging to start at 12:00 a.m. This requires some complexity in the PEV software to set up the requests at the time of connection. The basic data entry would define the target SOC and departure time (Time Charge Is Needed). The data entry for the first request could be for discharging at 5:00 p.m. at 2000 W for 2 hours. The recharge would be authorized to use time of use schedules (which have a 12:00 a.m. rate break). The PEV could wait until the discharge has been completed at 7:00 p.m. to calculate the parameters for the charging request and make it then. It could also estimate the SOC at the completion of the discharge and use that as the basis for providing the second request at the start of the session. This may require the PEV to update the request and cancel the earlier request after the discharge has been completed, but the controlling entity would have the benefit of having the stacked requests for planning.

It is also possible for the flow reservation server to provide a sequence of responses to a single request. This would allow, for example, a home energy management system to update the power available by providing a new response to the single request every few minutes to just offset demand. In this case, when the PEV connects to the EVSE at 5:00 p.m., the operator enables the PEV to discharge and engage with the home EMS for the home energy management application. Unlike the DER function, it is not possible when using flow reservation to switch between charging and discharging for a single request. The EMS can adjust the rate of discharging between zero and the requested power level for a discharge request, and it can adjust the rate of charging between zero and the requested power level for a charge request, but it cannot swing from discharging to charging. The DER function provides great flexibility for controlling power flows.

NOTE: The DER function uses a percent setpoint to command power flow. A positive value indicates a percent of the maximum discharge rate. A negative value indicates a percent of the maximum charging rate. This sign convention is the opposite of that used by flow reservation for energy request. The powerAvailable attribute in the flow reservation response is based on the DER function ActivePower object which provides for a signed integer value in Watts. The sign convention for ActivePower generally follows the DER convention. It is possible that a flow reservation server could provide an unsigned value, a signed value using the DER sign convention, or a signed value in the opposite convention. For this reason, it is recommended that the sign of powerAvailable not be used by the PEV for deciding between charging or discharging. If the energy request was for discharge, the value should be assumed as the discharge power. Unlike the DER Function, the flow reservation function is not allowed to swing power from charging to discharging, so there should be no confusion over always assuming the powerAvailable aligns with the energyRequested.

4.8 DER and Power Status End Device Resources

A DER device can be very complex, particularly so with a four quadrant converter that implements the autonomous curve functions. For an entity controlling a DER device to be able to use the DER device effectively, it must understand the capabilities and limitations of the device. SEP2 provides five resource objects that define the capability of the DER device. These five resource objects are shown as blue rectangles in Figure 13. Four resources are defined as part of the SEP2 DER function set. One resource, PowerStatus, is defined as part of the SEP2 power status function set. The PEVInfo Object, which was specifically added to SEP2 to support SAE J2836/3, is not an independently discoverable resource and is associated with the PowerStatus object. The WADL provides recommended URI structures for accessing each of these five resources. In this section guidance will be provided for how each of the attributes of these resource objects should be used with a PEV.

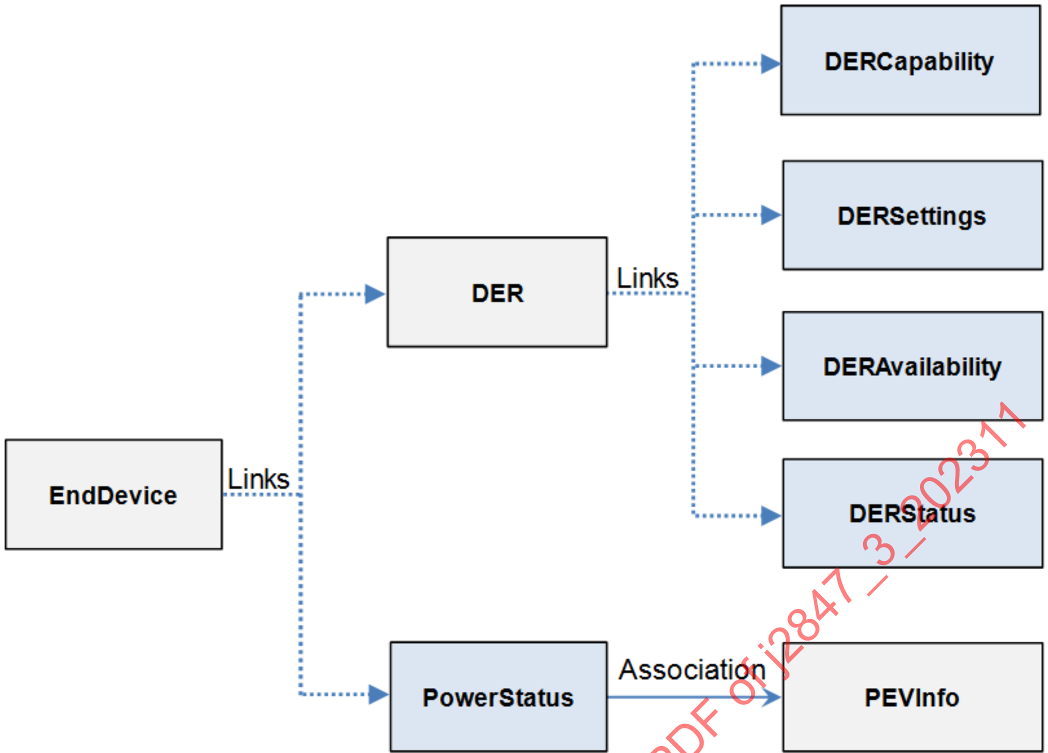


Figure 13 - State of the DER objects

4.8.1 Guidelines for the Object Attributes

SEP2 defines a model for transferring information. It does not define the process for creating the information. Much of the purpose of this section is to describe how information is entered into or created by the PEV. There is no single process and it all depends on the design of the PEV. Some fixed information can be constants in the embedded PEV software or settings that can be entered at manufacture, by a dealer, or by some other entity using a non-SEP2 exchange. This entry of information into a PEV by other than SEP2 communication is often called an "out of band" process. Other information can be the result of a calculation by the PEV using other more basic information that could be directly measured (such as a SAE J1772 control pilot setting), received using the SEP2 communication, entered by a driver using vehicle controls and displays, or some other method. In any case, the result of the calculation would be new information created by the PEV. The vehicle OEM must create the systems and software that are responsible for creating the values for the information which the PEV then shares using the SEP2 protocol.

The attributes of the DERCapability and DERSettings Objects are listed in Table 6. All of the attributes of the DERCapability object are fixed nameplate values and would not change during a charging session or even between sessions. Therefore, a DER controller would only need to request these values once during a session. For many reasons it is often desired to operate the inverter at lower levels than the nameplate values and these operational settings are provided by the DERSettings Object. The upper portion of the attribute section of the table matches each nameplate rating to a corresponding operational setting. Attributes which are unique to each object (and not matched) are shown in the lower portion of the attribute section of the table. Some guidance comments are shown in red below each attribute. Some will also be discussed in more detail below. For a PEV all elements of both objects must be considered to be read only values.

A DER controller should NEVER use any of the DERCapability attributes defined in the upper portion of the table that are matched with those for DERSettings. These should be used for information only, if at all with a PEV. The PEV must always provide a value for the corresponding DERSetting attribute. As will be discussed later, the EVSE control pilot can change the effective power rating during a session and there can be other factors that reduce the operational setting below the nameplate value.

The modesSupported attribute of the DERCapability object is used to define the inverter modes that are supported by the PEV. It uses a bit mask where individual bits define whether a mode is supported or not. The meaning for each bit location in the mask is defined by the DERControlType object in the SEP2 model. This is how a DER controller learns what functional capability can be provided by the PEV as a DER.

Table 6 - DER settings and capability objects

DERSettings Object Attributes	DERCapability Object Attributes
modesSupported (DERControlType)	modesSupported (DERControlType)
Set bit mask for supported functions	**Set bit mask for supported functions**
setMaxA attribute (CurrrentRMS)	rtgMaxA (CurrrentRMS)
Required	**Required**
setMaxAh attribute (AmpereHour)	rtgMaxAh (AmpereHour)
NOT Required for PEV (use setMaxWh)	**NOT Required for PEV (use rtgMaxWh)**
setMaxChargeRateVA attribute (ApparentPower)	rtgMaxChargeRateVA (ApparentPower)
NOT USED FOR PEV	**NOT USED FOR PEV**
setMaxChargeRateW attribute (ActivePower)	rtgMaxChargeRateW (ActivePower)
[U6 Maximum Forward Power]	**Nameplate Rating of Charger**
setMaxDischargeRateVA attribute (ApparentPower)	rtgMaxDischargeRateVA (ApparentPower)
NOT USED FOR PEV	**NOT USED FOR PEV**
setMaxDischargeRateW attribute (ActivePower)	rtgMaxDischargeRateW (ActivePower)
[U6 Maximum Reverse Power]	**Nameplate Rating of Inverter**
setMaxV attribute (VoltageRMS)	rtgMaxV (VoltageRMS)
NOT USED FOR PEV	**NOT USED FOR PEV**
setMaxVA attribute (ApparentPower)	rtgMaxVA (ApparentPower)
Not Required for PEV	**Not Required for PEV**
setMaxVar attribute (ReactivePower)	rtgMaxVar (ReactivePower)
[U7 Maximum Reactive Power], else 0	**Required for 4Q Converter, else 0**
setMaxVarNeg attribute (ReactivePower)	rtgMaxVarNeg (ReactivePower)
NOT USED FOR PEV DER	**NOT USED FOR PEV DER**
setMaxW attribute (ActivePower)	rtgMaxW (ActivePower)
SET TO Maximum Reverse Power (RQD)	**Required (Same as Discharge Rate)**
setMaxWh attribute (WattHour)	rtgMaxWh (WattHour)
Battery Capacity	**Nameplate Battery Capacity**
setMinPFOverExcitedPF attribute (PowerFactor)	rtgMinPFOverExcitedPF (PowerFactor)
[U7 Minimum Power Factor], else 1	**Required for 4Q Converter, else 1**
setMinPFUnderExcitedPF attribute (PowerFactor)	rtgMinPFUnderExcitedPF (PowerFactor)
NOT USED FOR PEV	**NOT USED FOR PEV DER**
setMinV attribute (VoltageRMS)	rtgMinV (VoltageRMS)
NOT USED FOR PEV	**NOT USED FOR PEV**
setVNom attribute (VoltageRMS)	rtgVNom (VoltageRMS)
Set by Out of Band Process	
SetESDelay attribute (Uint32)	rtgAbnormalCategory (Uint8)
NOT USED FOR PEV	**NOT USED FOR PEV**
SetESHIghFreq attribute (Uint16)	rtgNormalCategory (Uint8)
NOT USED FOR PEV	**NOT USED FOR PEV**
SetESHIghVolt attribute (Int16)	rtgOverExcitedPF (PowerFactor)
NOT USED FOR PEV	**Optional**
SetESLowFreq attribute (Uint16)	rtgOverExcitedW (ActivePopwer)
NOT USED FOR PEV	**Optional**

DERSettings Object Attributes	DERCapability Object Attributes
SetESLowVolt attribute (Int16) **NOT USED FOR PEV**	rtgReactiveSusceptance (ReactiveSusceptance) **NOT USED FOR PEV**
SetESRampTms attribute (UInt32) **NOT USED FOR PEV**	rtgUnderExcitedPF (PowerFactor) **Optional**
SetESRandomDelay attribute (UInt32) **NOT USED FOR PEV**	rtgUnderExcitedW (ActivePopwer) **Optional**
setGradW attribute (UInt16) **Set to 10000 (100% in 1.0 second)**	type (DERType) **Set to 81 (UInt8) for EV/EVSE**
setSoftGradW attribute (UInt16) **Set to 10000 (100% in 1.0 second)**	
setVRef attribute (VoltageRMS) **Set by Out of Band Process**	
setVRefOfs attribute (VoltageRMS) **Set by Out of Band Process**	
updatedTime attribute (TimeType) **Timestamp for update**	

Many of the attributes of the DERSettings object correspond with a parameter defined by an SAE J2836/3 use case. For example, setMaxDischargeRateW is equivalent to the Maximum Reverse Power in use case U6 (Basic DER). While the name may be different, the SAE J2836/3 document provides all of the information needed to understand the meaning and purpose of these attributes.

The values for the setMaxChargeRateW and setMaxDischargeRateW attributes of the DERSettings object are based on the lower of the limit established by the EVSE SAE J1772 control pilot setting and the effective inverter capacity. The effective capacity may be lower than the nameplate capacity based on the state of equalization of the battery cells, the age of the battery, the ambient temperature, and other factors. For most cases, the EVSE control pilot would be set at a fixed value. However, it is possible for it to be changed during a session and a DER control entity would need to request DERSettings regularly or subscribe to the resource.

If a four quadrant converter is provided in the PEV, the appropriate values for maximum reactive power and minimum power factor should be provided. Even though the attributes or the associated ratings and settings could be deleted from the XML payloads for inverters that do not support reactive power, it is recommended that default value of zero reactive power and unity power factor be provided with resource payloads. This provides some redundancy to the DER controller that the reactive power DER functions are not supported.

It is not possible for a PEV to know the reference voltage (setVRef) or the drop in voltage between the point of common coupling and the EVSE (setVRefOfs). These parameters are used for autonomous functions based on measured voltage. The values are specific to the premises and not the PEV. These are values needed by the PEV, but are not created by the PEV. They cannot be measured by the PEV. A measured line voltage of 225 VAC could be a low value for a 240 VAC reference or a high value for a 220 VAC reference. For a stationary DER, the site dependent values could be easily entered using an out of band process, but this is more of a problem for a roaming PEV.

The DERAvailability Object attributes are listed in Table 7 along with notes related to SAE J2836/3. The availabilityDuration and maxChargeChargeDuration apply to the full maximum charging and discharging power as defined by the DER Settings Object. For a PEV, the reserveChargePercent and reservePercent are defined to be 100%. The durations at maximum forward and reverse power change dynamically as the battery state of charge varies and as the time progresses during the session. These are two key parameters for the DER control entity to use to plan usage of the PEV.

Table 7 - DER availability object

DERAvailability Object Attributes
availabilityDuration attribute (UInt32)
[U6/U7 Duration Maximum Reverse Power]
maxChargeDuration attribute (UInt32)
[U6/U7 Duration Maximum Forward Power]
readingTime attribute (TimeType)
[U6/U7 Time of reference]
reserveChargePercent attribute (PerCent)
Set to 100% (10000)
reservePercent attribute (PerCent)
Set to 100% (10000)
statVarAvail attribute (ReactivePower)
[U7 Available Reactive Power]
statWAvail attribute (ActivePower)
[U6/U7 Maximum Reverse Power]

The DERStatus Object is shown in Table 8. Every attribute in this object is another object. Most of these objects consist of a dateTime object, which is itself a TimeType object, and a value. In most cases the value is an integer code. These are all explained in the SEP2 standard. The only required attributes are the stateOfChargeStatus and readingTime. The readingTime attribute is required to be implemented, but it is redundant with the dateTime provided by the stateOfChargeStatus object itself.

Table 8 - DER status object

DERStatus Object Attributes
alarmStatus (HexBinary32)
genConnectStatus (ConnectStatusType)
NOT USED FOR PEV DER
inverterStatus (InverterStatusType)
Optional
localControlModeStatus (LocalControlModeStatusType)
NOT USED FOR PEV DER
manufacturerStatus (ManufacturerStatusType)
Optional
operationalModeStatus (OperationalModeStatusType)
Optional
readingTime (TimeType)
Required
stateOfChargeStatus (StateOfChargeStatusType)
Actual State of Charge
storageModeStatus (StorageModeStatusType)
Optional
storConnectStatus (ConnectStatusType)
Optional

The attributes of the PowerStatus object are shown in Table 9. The power status function set was primarily established to track the state of small battery powered devices and these attributes are located in the upper half of the table. It can be used to learn when a device battery may need to be replaced. This was not intended to be used for a PEV traction battery. However, the estimated charge remaining can be used by a PEV to report state of charge. SOC is also available in the DERStatus object, but this value could be used if the DER function is not implemented in a PEV.

The attributes in the lower half of the table are part of the PEVInfo object which is associated with PowerStatus. The PEVInfo object is associated with the PowerStatus object but does not have its own WADL structure. When the PowerStatus object is requested from a PEV, the XML payload for the response will include the values for the PEVInfo object. This object has some information that looks similar to the attributes used with the flow reservation request object but care must be use when used with flow reservation. The energyRequestNow is always based on the charging requirement to move from the current SOC to the target SOC, even for cases where a flow reservation request energyRequested value may be negative. All of the attributes in PEVInfo are defined by SAE J2836/3 use case U6.

4.8.2 Special Considerations for SEP2

Many of the values for the attributes of these objects are dynamic. It is possible that a specific parameter, such as SOC, could be computed at a rate for internal use that is far higher than that needed by the DER controller for a specific V2G application. If these resource objects were updated in a host server for a client DER or sent to the DER controller in response to a subscription to the resource whenever the value changed internally, this could consume bandwidth for no good reason. The 2018 update of SEP2 provides a *postRate* attribute in the EndDevice resource for the DER controller to define the desired refresh rate for resource objects that could change values rapidly. The default rate for SEP2 is every 15 minutes which would be too slow for many V2G applications. Resouces that are not expected to change often (e.g., DERCapability) can be refreshed only on change, whereas resources that change rapidly are refreshed based on the *postRate* attribute.

Table 9 - Power status object

PowerStatus and PEVInfo Object Attributes
batteryStatus attribute (UInt8) **Set to 4 (Not applicable - Mains Powered)**
changedTime attribute (TimeType) **Required**
currentPowerSource attribute (PowerSourceType) **PowerSourceType (UInt8) = 1 (Mains)**
estimatedChargeRemaining attribute (PerCent) **Actual SOC (also in DERStatus)**
estimatedTimeRemaining attribute (UInt32) **NOT USED FOR PEV**
sessionTimeOnBattery attribute (UInt32) **NOT USED FOR PEV**
totalTimeOnBattery attribute (UInt32) **NOT USED FOR PEV**
PEVInfo Object (Associated with PowerStatus Object)
chargingPowerNow attribute (Active Power) **[U6 Active Power]**
energyRequestNow attribute (RealEnergy) **[U6/U7 Energy Request]**
maxForwardPower attribute (ActivePower) **[U6/U7 Maximum Forward Power - alternate]**
minimumChargingDuration attribute (UInt32)

PowerStatus and PEVInfo Object Attributes
[U6/U7 Minimum Charging Duration]
targetStateOfCharge attribute (PerCent)
Set by out of band process
timeChargeIsNeeded attribute (TimeType)
[U6/U7 Time Charge Is Needed]
timeChargingStatusPEV attribute (TimeType)
[U6/U7 Time of Reference]

All of the values for the attributes in these objects are read only. However, in some cases it is desired for a PEV to be able to accept a proposed value from another device using SEP2 rather than relying on some out of band process to acquire the information. For example, it would be useful to be able to get the value for setVRef from the EVSE. The PEV would need, of course, to validate the information using its own logic before accepting the value. If an EVSE included only the required resources for the SEP2 DER function set, even though it is not a DER, it could have a DERSettings resource with a setVRef value. A PEV could request the DERSettings from the EVSE and then use the value for its own setVRef. Similarly, the PowerStatus object could be used by an EVSE to provide access by a PEV to a proposed target SOC or revised Time Charge Is Needed.

It is beyond the scope of this document to define the use of SEP2 DER and power status function sets by an EVSE, for the purpose of providing proposed settings to a connected PEV. This is a design consideration for the EVSE. It is possible that an EVSE could use the DER function as a way for a home EMS to adjust the control pilot.

4.9 DER Programs

The PEV must enroll with a DER control entity to participate in a DER program. While a specific DER device receives a control signal (a DER Control) from the controlling entity, the same exact signal could be used by many other DER devices in the system. This is by design because it is often desired to manage a fleet of devices and have them all change output from 50 to 55%. A PEV client can participate in may DER programs. The DER Controller assigns the PEV client to a set of DER programs using the Function Set Assignment (FSA) resource. A DER program can represent the controls for a particular group. For example, a DER program can represent the controls for a particular feeder segment. All PEV clients connected to that feeder segment would participate in that program. A DER program can be assigned to a single client. In this case, there would be only one participant in that program. Using FSA's and DER Programs, the DER controller can target single clients as well as groups of clients.

NOTE: In this revision, Figure 14 has been removed. It is no longer needed, but the reference to this figure has been kept to maintain the numbering.

Figure 14 - Figure removed because it is no longer needed

4.10 DER Controls

The UCA® International Users Group published version 2.0 of the UCAIug Home Area Network System Requirements Specification in August 2010. This document guided the development of IEEE 2030.5-2018. Its requirements framework is segmented into five primary areas: HAN Applications, Communications, Security, Performance, and Operations-Maintenance-Logistics. HAN Application requirements are the heart of HAN interactions and define what HAN devices do. The requirements for HAN Applications are subdivided into four areas: Control, Measurement and Monitoring, Processes, and Human-to-machine interfaces. The Control HAN Application requirements are particularly relevant to the use of a PEV as a DER device.

Control HAN Applications respond to control signals, which are **Structured Messages** sent from an authorized party (e.g., Utility, service provider, or consumer EMS) requesting operational state change of a HAN device. This includes traditional direct load control commands and more advanced demand response commands where price or other data points may trigger a device to limit its energy consumption. It also includes control of reverse power flow and VAr for DER devices. Devices will respond within the operation of their control systems and algorithms. This response may be based on consumer preferences, internal safety systems, preconfigured thresholds, time-based values, and/or adaptive algorithms that may be present in the HAN device, EMS, and service provider solutions.

The control HAN Application requirements are restated here for a PEV receiving DER control signals:

- PEV shall accept DER control signals from one or more authorized parties (e.g., utility, service provider, or consumer EMS). [UCAI.App.Control.1]
- PEV shall adjust power flow as directed by a DER control signal receipt. [UCAI.App.Control.2]
- PEV shall resume previous operation state (as appropriate) following receipt of a DER control signal that cancels, expires, or overrides a previous DER control signal in effect. [UCAI.App.Control.3]
- PEV shall acknowledge receipt of a DER control signal when requested. [UCAI.App.Control.4]
- PEV shall acknowledge execution of a DER control signal when requested. [UCAI.App.Control.5]
- PEV shall acknowledge execution failure of a DER control signal. [UCAI.App.Control.6]
- PEV shall communicate any consumer or PEV initiated overrides or delays in response. [UCAI.App.Control.7]
- PEV shall implement its response to a DER control signal at a specified future time as set forth in the DER control signal. [UCAI.App.Control.8]
- PEV shall resume previous operational state (as appropriate) following expiration of a DER control signal. [UCAI.App.Control.9]
- PEV shall support randomization of scheduled DER control signal start and stop times to prevent unnecessary stress on the electric grid. [UCAI.App.Control.10] — **Only start randomization used by DER controller.**

A Structured Message is defined by the communication protocol as guided by an object model that describes functionality and data model associated with the HAN device. IEC 61850-7-420 defines the object model for inverter based DER devices and should be used by any protocol, including SEP2, for creating the Structured Messages for the control signals for DER devices. In SEP2 the Structured Message for controlling a DER is called a DER Control Object.

Table 10 provides details on the inheritance hierarchy of the DER Control Object. The DER Control is considered to be a Randomizable Event which is a member of the Event Class. All of the attributes of the hierarchy can apply to the DER Control. This means that the DER Control can have a start time, duration, randomized start, and randomized duration. If these are not specified for a DER Control, the current time should be used by the PEV as the start time with a default duration and zero used for the randomize attributes.

The Event Object is used more broadly in SEP2 than for only DER Controls. The nature of DER controls is such that Duration is not actually needed. In accordance with IEC 61850-7-420 and SAE J2836/3 the control signal has a reversion timeout, which is not the same as duration. It is expected that the DER controller will provide a new control signal to the DER before the reversion timeout. Consequently, this would normally be set for a longer time than the time of the expected update. The actual duration is established by the start of the next DER Control event. This attribute can be used to communicate the reversion timeout. Also, while SEP2 allows the randomize start attribute (time window in SAE J2836/3) to go from -3600 to +3600 seconds, for use with a DER a negative value should not be used and if is provided to the DER, it should be set to zero. Also, the randomize duration attribute should not be used for a DER and if a value is provided the DER should always use zero.

Table 10 - DER control signal

OBJECT	ATTRIBUTE
Resource **ResponableResource** **RespondableSubscribableIdentifiedObject**	
Event	creationTime attribute (TimeType) interval attribute (DateTimeInterval) start attribute (TimeType) **[U6/U7 Start Time]** duration attribute (UInt32) **[U6/U7 Reversion Timeout]** eventStatus currentStatus (UInt8) dateTime (TimeType) potentiallySuperseded (Boolean) potentiallySupersededTime (TimeType) Reason (String192)
RandomizableEvent	randomizeStart attribute (OneHourRangeType) **[U6/U7 Time Window; value ≥ 0]** randomizeDuration attribute (OneHourRangeType) **NOT USED FOR PEV DER**
DERControl Object	DERControlBase Object () opModConnect (Boolean) opModEnergize (Boolean) opModFixedPFAbsorbW (PowerFactorWithExcitation) opModFixedPFInjectW (PowerFactorWithExcitation) opModFixedVar (FixedVar) opModFixedW (SignedPercent) **[U6 SetPoint)** opModFreqDroop (FreqDroopType) opModFreqWatt (DERCurveLink) opModHFRTMayTrip (DERCurveLink) opModHFRTMustTrip (DERCurveLink) opModHVRTMayTrip (DERCurveLink) opmodHVRTMomentaryCessation (DERCurveLink) opModHVRTMustTrip (DERCurveLink) opModLFRTMayTrip (DERCurveLink) opModLFRTMustTrip (DERCurveLink) opModLVRTMayTrip (DERCurveLink) opModLVRTMomentaryCessation (DERCurveLink) opModLVRTMustTrip (DERCurveLink) opModMaxLimW (PerCent) opModTargetVar (ReactivePower) opModTargetW (ActivePower) opModVoltVar (DERCurveLink) opModVoltWatt (DERCurveLink) opModWattPF (DERCurveLink) opModWattVar (DERCurveLink) rampTms (UInt16) **[U6/U7 Ramp Time]**

Table 11 - DER control function commands

DER Function			Attributes of DERControlBase Object	
Category	**Type**	**Name**	**Name**	**Object**
Active	Fixed	Fixed Watt	opModFixedW	SignedPerCent
	Fixed	Max Limit Watt	opModMaxLimW	PerCent
	Fixed	Target Watt	opModTargetW	ActivePower
	Curve	Frequency-Watt	opModFreqWatt	DERCurveLink
	Curve	Volt-Watt	opModVoltWatt	DERCurveLink
	Curve	Frequency-Droop	opModFreqDroop	FreqDroopType
Reactive	Fixed	Fixed Power Factor Absorb	opModFixedPFAbsorbW	FixedPowerFactorWithExcitation
	Fixed	Fixed Power Factor Inject	opModFixedPFInjectW	FixedPowerFactorWithExcitation
	Fixed	Fixed Var	opModFixedVar	FixedVar
	Fixed	Target Var	opModTargetVar	ReactivePower
	Curve	Volt-Var	opModVoltVar	DERCurveLink
	Curve	Watt-Power Factor	opModWattPF	DERCurveLink
	Curve	Watt Var	opModWattVar	DERCurveLink
LVRT	Curve	Low Voltage Ride Through May Trip	opModLVRTMayTrip	DERCurveLink
	Curve	Low Voltage Ride Through Momentary Cessation	opModLVRTMomentary Cessation	DERCurveLink
	Curve	Low Voltage Ride Through Must Trip	opModLVRTMustTrip	DERCurveLink
	DO NOT USE IN PEV AT THIS TIME			
HVRT	Curve	High Voltage Ride Through May Trip	opModHVRTMayTrip	DERCurveLink
	Curve	High Voltage Ride Through Momentary Cessation	opModHVRTMomentary Cessation	DERCurveLink
	Curve	High Voltage Ride Through Must Trip	opModHVRTMustTrip	DERCurveLink
	DO NOT USE IN PEV AT THIS TIME			

The DERControlBase Object which is part of the DERControl incorporates the attributes (or commands) for each of the inverter modes listed in Table 11. It is not required that the DER inverter implements each function and there is a provision in SEP2 for the DER to define which functions are supported. Because a DERControl cancels the current DERControl when it starts, all of the functions being used must have values provided even if not changed from the current control. If a DERControl is provided to change the setpoint for fixed power factor, the setpoint for fixed flow must also be provided even if it is not changed. This is the expected behavior for the DER controller entity. The controller should also not issue more than one command for each band in the table: active, reactive, LVRT, and HVRT. Only one method at a time can be used by the inverter to control active and reactive power. The SEP2 protocol will allow conflicting values to be sent and received, so it is up to the inverter software to resolve conflicts. It is recommended that the priority for conflict resolution be a listed in the table, with fixed flow and fixed power factor being the highest priority if there is a conflict. This conflict will be presented by a single controller, which should not allow this to happen. Conflicts between DER control signals received from two different control entities are a different problem.

The low and high voltage ride through functions should not be implemented in a PEV at this time. This will be discussed in more detail in 4.13.

Many of the attributes use another object as the basis for defining the command value. For the opModFixedW attribute, the SignedPerCent object is used. This is a signed integer value of 100 times the percent of the maximum charging or discharging power. A negative sign is used for charging. Even though reverse flow seems like it should be a negative value, conventions for active power produced by a DER device require that to be positive. This means discharging is positive and charging is negative. To set active power of the DER to 0.503 times the maximum charging power, the opModFixedW would be set to -5030 (-50.3%). The XML would look like:

```
<opModFixedW> -5030 </opModFixedW>
```

The SEP2 model must be studied to understand the definition of the information and how it must be decoded for internal use in the inverter. The SEP2 protocol can fetch the integer -5030 value from the DER controlling entity, but to be used in the embedded inverter software it may need to be moved and converted from the integer value of -5030 to a floating point value of -0.503 in another memory location. A primary purpose of this document is to help guide the development of the PEV software to actually move the XML type information into a useful form and location in memory for the actual PEV software.

4.11 Event Rules and Guidelines

The SEP2 Application Specification provides rules and guidelines for Events. These rules are not part of the SEP2 communication model, but define decision requirements that would be implemented in the device software. These rules primarily relate resolving potential conflicts caused by overlapping events, unintended gaps between events, or other anomalous behavior. The SEP2 communications present the conflict to the device and it is the device software that uses the rules to resolve the conflicts.

4.11.1 DER Control Signals as Events

In the SEP2 model, a DER Control is considered to be an event. It is also a special type of event called a randomizable event. Many types of events are defined by SEP2 and the rules are not specific for DER control events and DER devices. However, SEP2 does provide some distinction for function sets with direct control (such as DER, DRLC, and flow reservation) versus informational function sets. The DER control object and its inheritance hierarchy in the SEP2 Object Model were discussed in 4.10. While that section was primarily focused on the control modes for a specific event, the UCAIug HAN SRS requirements are directly relevant to the rules of engagement for a DER device.

Figure 15 shows the structure of a DER event as defined by EPRI Common Functions and SAE J2836/3 for a real power setpoint mode (opModFixedW). The control signal arrives at the DER at a current time and the will have a defined Start Time, a randomization time window, a ramp time, a reversion timeout, and a setpoint value. The structure is similar for all for the other control modes.

Time Window must always represents a time delay. In the SEP2 model, Time Window is provided by the randomizeStart attribute of the RandomizableEvent Object. SEP2 allows the randomizeStart attribute to go from -3600 to +3600 seconds. However, if a negative value for randomizeStart is received by a DER device, it should internally set the value to zero before using it. The setpoint from the currently active control is maintained until the start time plus time window multiplied by a uniform random number (0 to 1). This is the effective start time for starting to linearly transition between the current setpoint value and the new setpoint value over the Ramp Time, which is provided as an attribute in the DERControlBase Object.

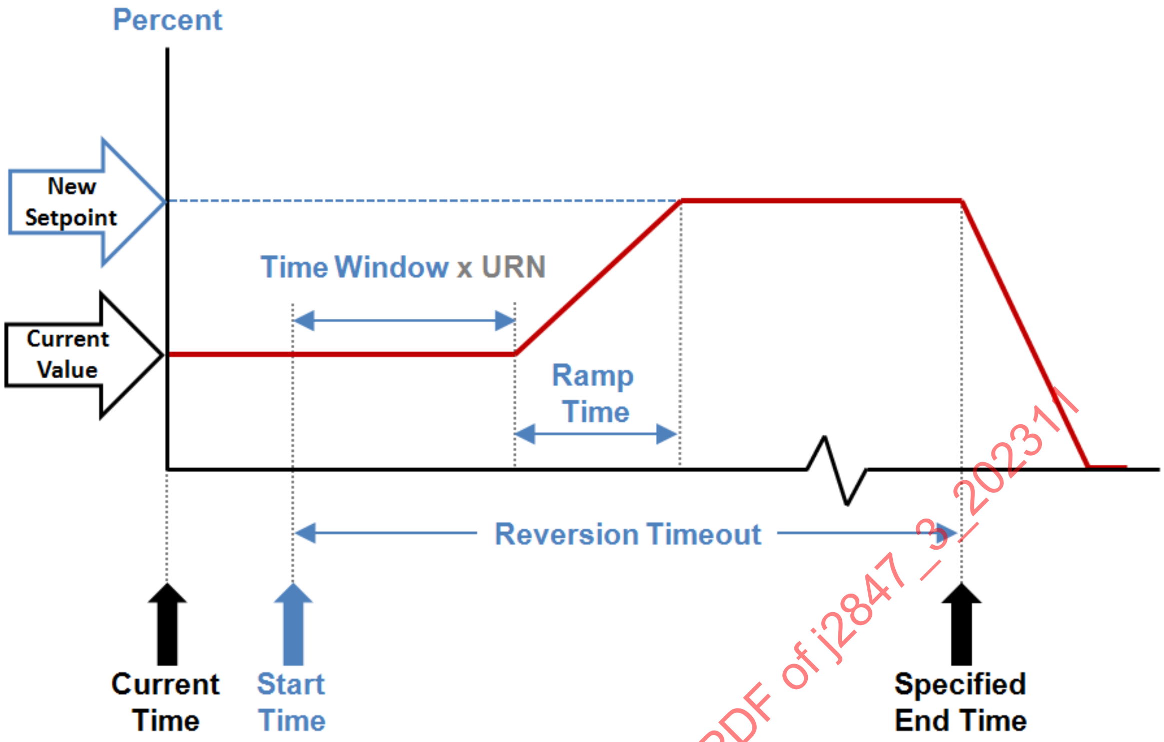


Figure 15 - DER control event

The reversion timeout shown in the figure for the DER control signal looks like an event duration. And if only a single DER control signal was provided during a DER application session, it would be the duration of the one DER event. A DER controller is responsible for always ensuring that a valid DER Control is presented to the DER device and that there are no gaps between control events. The reversion time is not an actual desired duration, except for possibly the last expected control for a session. Even the last control provided by the DER controller during a session would normally be expected to be a managed ramp to zero active and reactive power. The DER controller can end a session by ramping to zero and not providing a control before the timeout. This disengages the DER session. It is expected and required that the DER controller provide a DER Control to the DER device with a start time that overlaps the active control. The failure by a DER controller to provide a continuous sequence of overlapping control signals should result in a disengagement from the DER program and reversion by the PEV to a normal operational state of the device.

The reversion timeout is based on the expected start time of the next event. It should be long enough to allow for variations in the expected rate and transport delays, but not so long that the DER device continues to execute the last command if the DER controller or communication system has failed. It is the “dead man’s” switch for the DER application. In an application where commands are routinely provided every 60 seconds, a reversion timeout of 120 seconds might be reasonable. If power is to be maintained flat for 30 minutes, the controlling entity would send out a command to the DER every minute; if it were 60 seconds late, the DER device would revert to its default condition. However, the controlling entity could provide a reversion timeout of 31 minutes if it did not expect to issue a new control for 30 minutes, even though it could issue them every minute. The 31 minute reversion keeps the same 60 second dead man interval.

Because of the use of the duration as form of dead man’s switch, it should not be randomized. It needs to be absolutely predictable. The controlling entity knows exactly the start time and reversion time of the current control. The specified end time is known. The next event must be received by the DER device and have a start time before the specified end time. This is known to both the DER controller and the DER device. If there is a gap, it is either a failure condition or the DER controller is signaling the end of the session. The randomizeDuration attribute should not be provided to a DER device by the controlling entity and if a value is provided, the DER device should set the value to zero for internal use.

4.11.2 SEP2 Event Rules and Guidelines

The SEP2 standard includes a major section called “Common Functionality” (refer to 12.1). It is primarily focused at events and event conflicts. Table 12 provides a list of event definitions from SEP2. The basic term is shown in the left column and the definition is provided in the right column. The top section shows the parameters are part of the Event Object. The section below that provides definitions for certain terms that are simply calculated from the basic event attributes. The lower portion defines terms are not easily calculated but are based on the relationship between events. All of these terms are used throughout a section titled “Event Rules and Guidelines” in the SEP2 standard. Where SAE J2836/3 uses a different term, it is shown in red text below the SEP2 term. The SAE definition is also shown in red. In some cases, because of restrictions for use with a DER, versus other events, this results in some simplifications. For example, a DER event does not allow start time randomization to be earlier than the defined start time. The duration, because it is a timeout, is also not intended to allow randomization. For a PEV as a DER, the red definitions should be used when interpreting the SEP2 rules of engagement for events.

The SEP2 rules and guidelines should be followed for DER events, flow reservation response events and other event based operations.

4.11.3 SEP2 Default DER Control

The SEP2 specification states that “if there are no active Events, the DER device SHALL be managed by the Default DER Control instance exposed by the preferred DER Program.” The UCAIug HAN SRS requires that a DER shall resume previous operation state (as appropriate) following expiration of a DER control signal. These are not equivalent and SEP2 is not correct.

It does not make sense for the entity that failed to execute its control function to determine the desired default behavior as gap filler. If a DER control times out, this is a system failure, and the PEV should revert to normal operational state. For a PEV, it could switch to a flow reservation or price based charging or even unrestricted charging—none of which are based on the DER functions. The embedded software of the device should determine its own behavior after disconnect.

It is recommended that the Default DER Control not be implemented for PEVs.

Table 12 - SEP2 event definitions

Term	Definition
Provided with DER Control (EVENT)	
Start Time	Event: Interval: Start (UTC in seconds from 12:00 on 1/1/1970)
Duration **(Reversion Timeout)**	Event: Interval: Duration (in seconds) **Similar but not an actual intended duration**
Start Randomization **(Time Window)**	RandomizableEvent: randomizeStart (in seconds) **Similar but must never be a negative value**
Duration Randomization **(Not Used for DER)**	RandomizableEvent: randomizeDuration (in seconds) **Not used or must be set to zero**
Calculated Value	
Effective Start Time	Start Time + (Start Randomization x Random Number) **Start Time + (Time Window x Random Number)**
Earliest Effective Start Time	Earlier of Start Time or Effective Start Time **Equals Start Time**
Effective Duration	Duration + (Duration Randomization x Random Number) **Equals Reversion Timeout**
Specified End Time	Start Time + Duration **Start Time + Reversion Timeout**
Effective End Time	Effective Start Time + Effective Duration **Equals Specified End Time**
Scheduled Period	Paired (Start Time, Specified End Time)
Effective Scheduled Period	Paired (Effective Start Time, Effective End Time) **Paired (Effective Start Time, Specified End Time)**
Test Condition	
Overlapping Event	Start Time earlier than Effective End Time of earlier event
Successive Event	Start Time equal to Specified End Time of earlier event
Nested Events	Start Time > Start Time of earlier event AND Scheduled End Time < Scheduled End Time of earlier event

4.11.4 Establishing the Update Rate for DER Controls

In SEP2, the client DER device is responsible for requesting new DER Control instances from the entity controlling the DER devices. The WADL shows the following structure for requesting a specific DER Control instance from the controlling entity:

> **DERControlList Resource**
> List of DERControls.
>
> Sample URI: /derp/{id1}/derc
> Request Representation: DERControl
> Response Representation: DERControlList
> Methods: GET/HEAD: Mandatory, PUT: Error, POST: Optional, DELETE: Error

SEP2 guidelines state that a DER device that does not subscribe to the DERControList shall poll the lists for new DERControls at least once every 15 minutes and should poll at least every 5 minutes. This would require that any DERControls would have a reversion timeout of at least 15 minutes. This type of polling would not be useful for applications that required tighter coordination such as frequency regulation with command potentially being produced by the controlling entity every few seconds. Even energy management in a home or demand charge management in a business could require updates at least every 1 minute.

SEP2 does provide a means for the DER controller to communicate the desired polling rate for the client DER to use. The pollRate attribute is available in the top-level resource for every function set to define the default polling rate for the function set. For the DER function set, the DERProgramList resource contains the *pollRate* attribute. Althought the name of the attirbute is *pollRate*, it is actually a polling interval with units of seconds. The client DER polls the resources of this function set (including the DERControlList) every *pollRate* seconds.

The polling interval must not only consider the update rate of the DER control signals, but it must also consider the transport delays. It does not have much value to poll every 4 seconds for a scheduled update every 4 seconds if the request just misses the update. This adds 4 seconds to the transport delay. If the DER controller and DER device are not tightly synchronized, the client DER may have to poll every second, for example. This reduces the potential time delay from 4 to 1 second.

The DERControlLists are subscribable resources. If the DER device subscribes to the list of the controlling entity, the controlling entity will provide a notification when the list changes which could even include the XML for a new DER control. However, SEP2 recommends that subscription not be used for updates more frequently than every 30 seconds. This might be acceptable for use in facility demand charge management at 1 minute or 2 minute update rates, but not for frequency regulation where new commands could arrive every few seconds. The subscription process may be a preferred approach for frequency regulation. The subscription and notification process are discussed in detail in the SEP2 standard.

4.12 Curve Functions

SEP2 defines five autonomous functions: frequency-watt, volt-var, volt-watt, watt-power factor, and watt-var. These along with voltage and frequency ride-through curves are considered to be curve functions in SEP2 because they are defined by a sequence of data points as an X-Y curve. The frequency and voltage ride-through curves will be discussed in a subsequent section. This section only discusses the five autonomous functions. The purpose, structure, and operation of these functions are described in the EPRI Common Functions report and in SAE J2836/3, so this section will focus primarily on how the information describing the functions is provided to the DER device using SEP2 and issues associated with how the DER device prepares the functions for use.

The message structure used to transfer the definition of each curve function from the DER controller to the DER device is shown in Table 13. The DER curve is an identified object which is in turn is a resource. When a PEV GETs a DER Curve from the DER controller the responding XML message could include all of the attributes in the table. Up to ten pairs of x-y data points would be included as part of the definition. The order of the data points is significant and the pairs cannot just be sorted in the order of x-values. The curve is linked to a specific DER program of the DER controller.

Table 13 - DER curve functions

OBJECT	ATTRIBUTE
Resource	href attribute (anyURI)
IdentifiedObject	description attribute (String32) mRID attribute (mRIDType) version attribute (VersionType)
DERCurve **(Linked to DERProgram)**	autonomousVRefEnable attribute (boolean) **Optional** autonomousVRefTimeConstant attribute (UInt32) **Optional** creation time attribute (TimeType) curveType attribute (DERCuveType) openLoopTms attribute (UInt16) **Optional** rampDecs attribute (UInt16) **Optional** rampIncs attribute (UInt16) **Optional** rampPT1s attribute (UInt16) **Optional** vRef attribute (PerCent) **Optional** xMultiplier attribute (PowertOfTenMultiplierType) yMultiplier attribute (PowertOfTenMultiplierType) yRefType attribute (DERUnitRefType) **CurveData Object (Up to 10 pairs)** excitation attribute (boolean) xvalue (Int32) yvalue (Int32)

The curve type is defined by the curveType attribute. Each function is assigned a numerical code: frequency-watt (0), volt-var (11), watt-power factor (13), volt-watt (12), and watt-var (14). The XML defining the curve type for a frequency-watt curve would be:

```
<DERCurve>
    …
  <curveType>0</curveType>
    …
</DERCurve>
```

The yRefType attribute defines the context of the y-axis using numerical codes defined by the DERUnitRefType Object. Examples of yRefType values are: %setMaxW (1) %setMaxVar (2) and %statVarAvail (3).

There is some redundancy and potential for conflict in the identification of curve types. The curve that is requested from the DER controller by the PEV and provided in the response has a defined type embedded in the XML payload. This is important for processing the data and setting up the data points with the appropriate independent and dependent variables. This is done by the PEV operational software. This only prepares a function to be used.

A function is activated using the DER Controls discussed in 4.10. Each curve is activated by providing the link to the function as an attribute in the DERControlBase object. A specific volt-var function, of which several could be implemented in the PEV, is activated by:

```
<DERControl>
     <DERControlBase>
     …
          <opModVoltVar>
                    href = URI (for a specific volt-var curve)
         </opModVoltVar>
     …
     </DERControlBase>
</DERControl>
```

The URI reference is to the function that could have already been loaded from the DER controller and implemented in the PEV operational software. This URI link to the external location needs to somehow activate the corresponding function in the operational software. During the loading and processing of the curves, the association between the URI and the actual operational function to be activated needs to be made.

However, the function may or may not actually be a volt-var function. The function is defined when it is loaded, not by the URI. There is an implicit assumption in the SEP2 specification that the <opModVoltVar> link actually points to a volt-var curve. A mis-configured server could have the <opModVoltVar> link point to a freq-watt curve. The PEV can check this by verifying the curveType attribute of the actual downloaded curve matches the intended type in the DER Control link. If the <opModVoltVar> calls a non-volt-var function, this should result in an error. The stored function will always be correct in the PEV. If the DER Control calls up a function that has not been previously loaded, this command should prompt the PEV to fetch the curve and prepare it for use.

Curves belong to the DER controller but are implemented in the PEV. Great care must be exercised if the PEV is allowed to store functions between sessions. This might be fine, if the PEV reengages with the DER controller at the same EVSE. But if it engaged on a different EVSE on a different distribution feeder, the DER controller might require different curves. This may not be a problem if each curve had a unique identification at the DER controller. One option is for a PEV to purge curves after a session closes and require all curves to be loaded at the start of a session or when a DER Control first activates one that has not been loaded.

SEP2 is concerned about providing the capability for the DER device to download the parameters and data points that describe a curve function from the DER controller for the application. The PEV must actually implement operational functions from the received information. This is not a trivial problem. The information needs to be checked to verify that it will actually work with the PEV software. It cannot just be assumed that the table points are well designed. This can only be verified by understanding the actual PEV algorithms used to implement the functions. What may work with a floating point implementation may overflow with a fixed point implementation. The PEV could use predefined functions that data points are placed into depending on the curve characteristics or it could even create software around the data. There are many ways to create the executable functions, but any method will require some data analysis.

A series of examples of curve data sets and functions will be shown. Assume that the range of the x-axis independent variable is from zero to ten. Figure 16 shows a normal piecewise continuous function. The data points are transmitted in the order (0, 3), (4, 2), (7, -2), and (10, -3). If the PEV has a linear table lookup function, y=f(x), these points could be directly used to execute this function. Care would need to be used if the slope of any line segment could result in an arithmetic overflow. The XML sequence would look like:

```
<DERCurve>
      …
      <xMultiplier>0</xMultiplier>
      <yMultiplier>0</yMultiplier>
      <xvalue>0</xvalue>
      <yvalue>3</yvalue>
      <xvalue>4</xvalue>
      <yvalue>2</yvalue>
      <xvalue>7</xvalue>
      <yvalue>-2</yvalue>
      <xvalue>10</xvalue>
      <yvalue>-3</yvalue>
      …
</DERCurve>
```

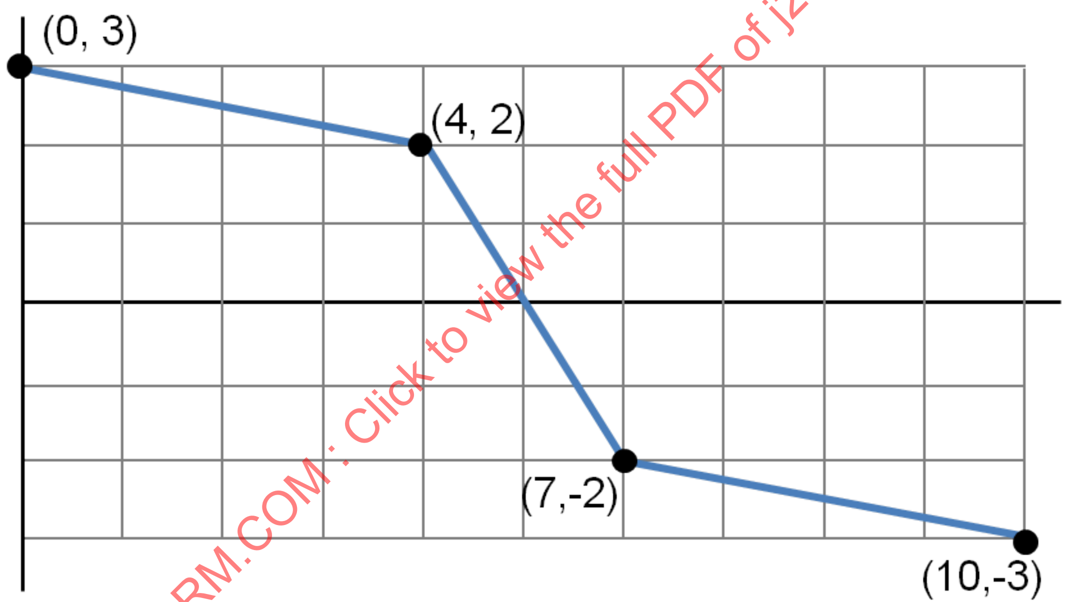


Figure 16 - Normal piecewise linear curve

Figure 17 shows a curve that is not a mathematical function because it has more than one value for a given x value. It is a non-convex curve and indeterminate. However, it could be sent using the values (0, 3), (6, 2), (4, -2), and (10, -3). The reverse slope allows three values for any x between 4 and 6. Depending on how a table lookup function performed its search it might even execute using first and third segments and jump back. But this is a bad curve.

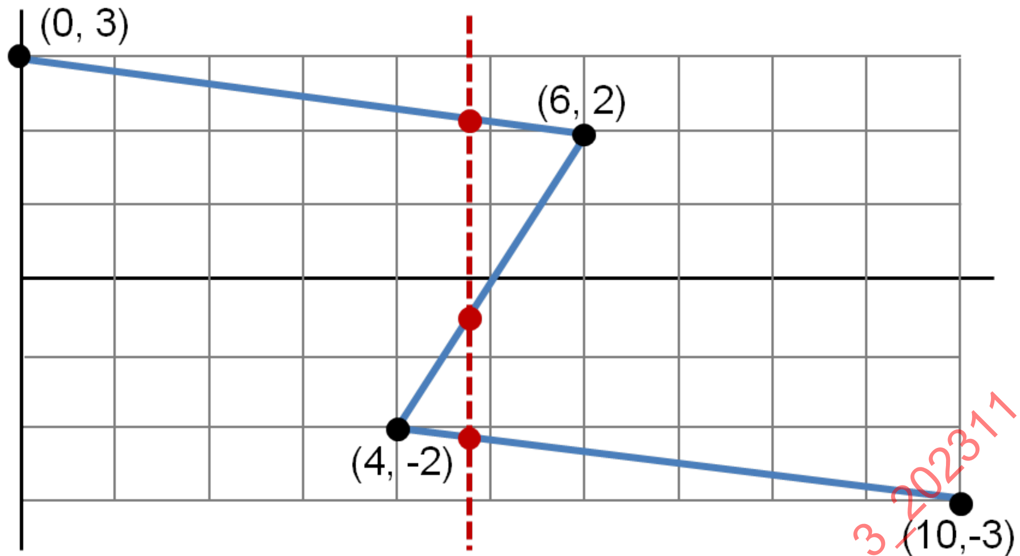


Figure 17 - Abnormal piecewise linear curve

A discontinuous function is shown in Figure 18. This function is commonly used, sometimes with hysteresis, but it will fail if used with a linear table lookup function because of the indeterminate vertical line segment. These points could be received and the PEV could create a function that simply executed this IF statement:

If X < 5, then y = 2, else y = -2

But this requires creating software for a function. It could fail as a table, if the lookup function did not explicitly allow for this type of discontinuity. If the points (0, 2), (5, 2), (5, -2), and (10, -2) were received this must be either flagged as an error or the PEV must create a discontinuous function.

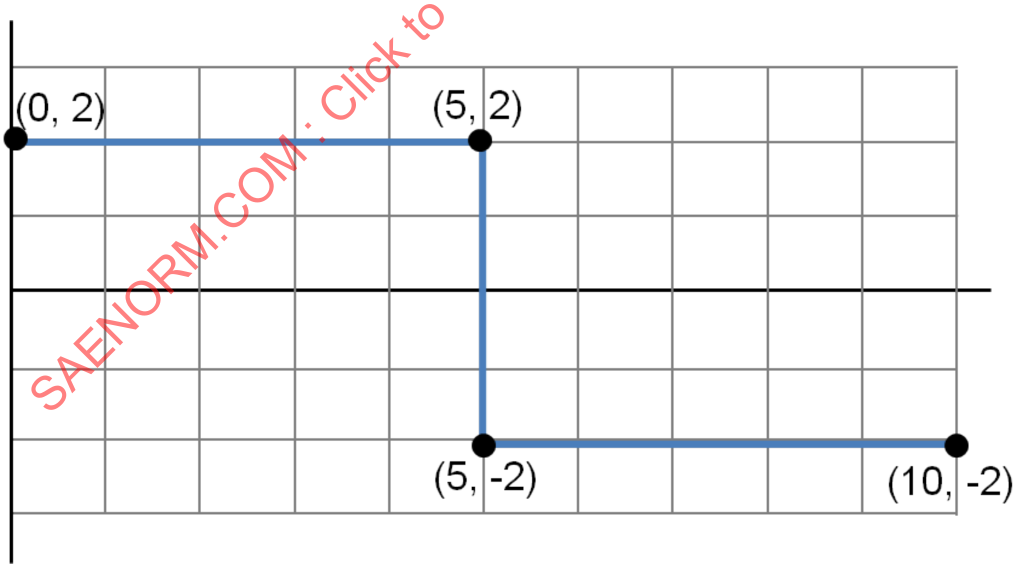


Figure 18 - Discontinuous function

Figure 19 shows an example of hysteresis. This shows the importance of the order of the XML sequence. Only four points are actually needed to define the function: (5, 2), (7, -2), (5, -2), and (3, 2). The logic is supposed to carry the first point to the lowest x value as a flat segment and the furthest x value to the maximum x value as a flat. In an actual table, it could require adding the data to the PEV embedded software. The internal function is best described by (0, 2), (5, 2), (7, -2), (10, -2), (5, -2), (3, 2), and (0, 2). Because of the hysteresis, this is not an actual function that will cleanly fit a linear table lookup subroutine. This function could be used and is allowed. The hysteresis needs to be recognized and distinguished from a non-convex or discontinuous curve. There is a forward curve of (0, 2), (5, 2), (7, -2), and (10, -2) and a reverse curve of (0, 2), (3, 2), (5, -2), and (10, -2) and hysteresis logic. There is no separate designation of hysteresis so the PEV must recognize the pattern and configure the function appropriately.

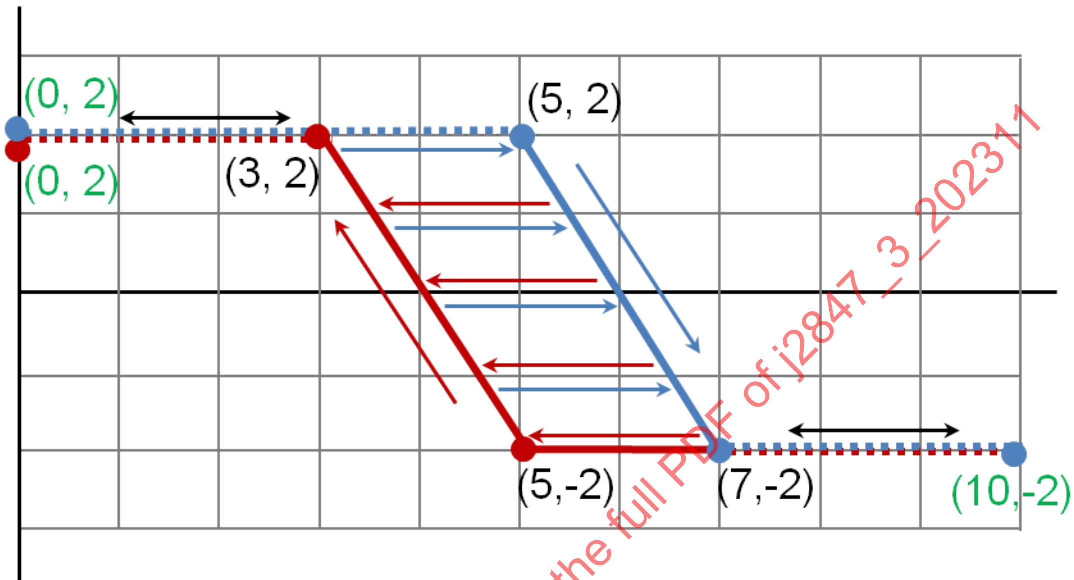


Figure 19 - Curve function with hysteresis

There is great excitement in the electric power industry about the future ability of solar PV systems to implement advanced inverter functions, such as volt-var curves with hysteresis. But no one has figured out how to pay the solar PV system owner for providing the service. This could be made a condition of participation in net-metering for the larger producers. It is much more questionable as to whether a PEV owner could be compensated enough for providing volt-var service to justify the complexity of installing a four-quadrant inverter and implementing the autonomous functions. SEP2 will provide the capability to transfer the curve data into the PEV and to activate the function. But much of the complexity will be in the inverter and the associated operational software to actually automatically validate and then configure the functions and operate them.

4.13 Voltage and Frequency Ride Through Functions

A grid-connected inverter must not supply power into a failed grid for the safety of people near any downed lines and repair personnel. One test for a failed grid is that the line voltage or frequency goes out of normal operating limits and the inverter must shutdown within a specified maximum duration that depends on the magnitude of the voltage or frequency transient. IEEE 1547-2018 establishes voltage and frequency limits that are generally followed in North America.

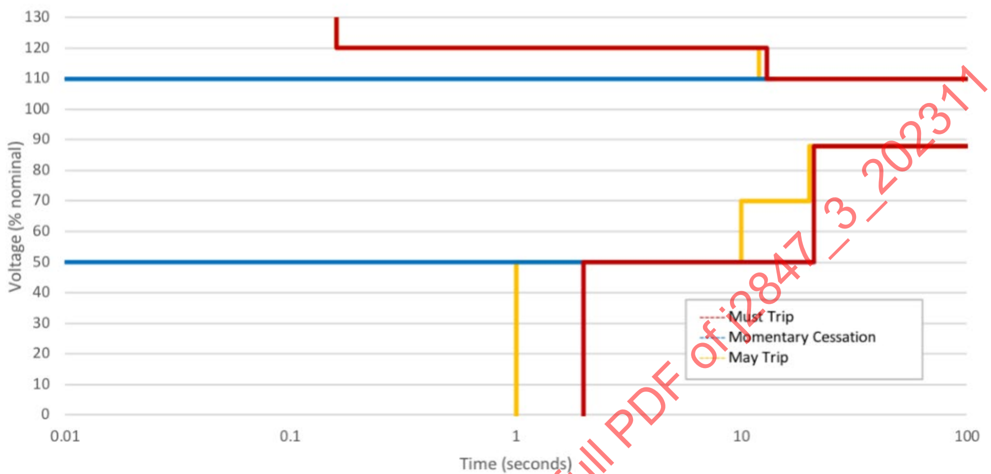
SEP2 uses Must Trip, May Trip, and Momentary Cessation curves to define regions associated with voltage and frequency must trip, may trip, and momentary cessation behavior. Each region is defined with a piece-wise linear curve demarcating the boundary (e.g. when crossing curve, the inverter is in the may trip region). The independent axis (x-axis) of these curves in time **duration**, not time—that is, the x-axis represents the length in time of the disturbance in seconds, not the time when the disturbance occurred.

When the inverter enters the must trip region, the inverter must trip. When the inverter enters the may trip region, the inverter may continue in its current operational mode (either cease to energize or mandatory operation) or may trip. When the inverter enters the momentary cessation region, the inverter shall cease to energize but shall not trip.

The Must Trip curve has the highest precedence, followed by the Momentary Cessation curve, followed by the May Trip curve having the lowest precedence. When crossing a curve with higher precedence, the inverter shall assume the behavior of the higher precedence curve.

Curves are assumed to extend indefinately in the vertical direction from the first point of the curve (i.e., the point with the smallest time duration value). Curves are assumed to extend indefinately in the horizontal dirction from the last point of the curve (i.e., the point with the largest time duration value).

An example set of voltage ride through curves is shown in Figure 20. The x-axis of time duration uses a log scale. The points defining the various voltage ride through curves are shown below the graph.



Curve	Points
LV must trip	(2, 0), (2, 50), (21, 50), (21, 88), (100, 88)
LV momentary cessation	(0, 50), (15, 50)
LV may trip	(1, 0), (1, 50), (10, 50), (10, 70), (20, 70), (20, 88), (100, 88)
HV must trip	(0.16, 130), (0.16, 120), (13, 120), (13, 110), (100, 110)
HV momentary cessation	(0, 110), (13, 110)
HV may trip	(0.16, 130), (0.16, 120), (12, 120), (12, 110), (100, 110)

Figure 20 - Voltage ride through function example

Using this example, if the voltage drops below 88% for more than 21 seconds, the inverter must trip. If the voltage drops below 50% for more than 2 seconds, the inverter must trip. If the voltage increases above 120% for more than 160 ms, the inverter must trip. If the voltage increases above 110% for more than 13 seconds, the inverter must trip.

Ride through curves are typically implemented in hardware in the inverter. Curve points are programmed into the inverter and the inverter interprets the defined curves based on the rules described above. Although it is theoretically possible to implement ride through curves in PEV software, this is not recommended. First, ride through curves are used for grid safety. Software implemenations may not provide enough reliability for this critical function. Second, some Must Trip times are on the order of 100 ms. Software implementations may not be able to guarantee response times this fast.

4.14 Application Examples

This section provides a brief overview of a few V2G application examples. The details for each example will be provided in a separate appendix. The focus of each example is the interaction between a single PEV and the controlling entity for the V2G application. The communication of the controlling entity with other systems is not relevant—except as in comments regarding how it determines what to send or receive from the specific PEV.

4.14.1 Home Energy Management Using Flow Reservation (Appendix B)

A homeowner plans to use the available energy in a PEV to offset residential loads during a period of higher energy prices in the evening and then recharge the PEV battery at lower rates after midnight. A home energy management system will be used to control the rate of discharging of the PEV to offset other residential loads and maintain the total premises demand below a target level. While the home EMS and PEV could perform this application using the fixed flow mode of the DER function set, this example is based on using the flow reservation function set.

4.14.2 Frequency Regulation (Appendix C)

One of the most talked about applications for V2G is the concept of an aggregator enrolling hundreds of PEVs to perform frequency regulation for the bulk grid. The aggregator bids this ancillary service to the system operator as a block of capacity for a specific hour in the future. During the awarded bid interval, the aggregator receives an automatic generation control (AGC) signal from the system operator every few seconds and in turn adjusts the rate of charging or discharging for each PEV in this large fleet to perform the service. The PEV owners are compensated by the aggregator for participating in the program. This example is based on using the fixed flow mode of the SEP2 DER function set.

5. AC V2H/V2G SUMMARY

5.1 V2G-AC - EV Requirements

SAE J3072 is published as an SAE Standard and defines vehicle requirements for V2G-AC. This is a primary document to follow when creating and deploying any V2G-AC system.

SAE J2836/3 is published as an SAE Information Report and defines "use cases" which consist of one or more specific DER functions. Use Case U6 is identified as "Basic DER Function" which defines the control of active power to a setpoint provided by a controlling entity. Use Case U7 is identified as "Advanced DER Functions" and consists of a bundle of individual DER functions which may be required to be implemented by a DER.

Generally, the term "Use Case" is used to describe an end-to-end purpose. In SAE J2836/3, the term "V2G Application" was used to describe how the EV-EVSE performs certain functions for the benefit of the utility power system, the distribution feeder, and/or a behind the meter customer. Table 14 is also in SAE J2836/3 and identifies V2G applications. This is not an exhaustive list of applications. The ability to control active power (U6) can be used in many different ways. For example, U6 can be used by a utility to control transformer load. This is different than using it to manage demand behind the meter. In either case, it only requires commanding active power.

Additional requirements: The EV is not required to lock the EVSE connector for AC charging and is also not required for AC V2G. The EV is to power down within 100 ms when the S3 switch is pressed as identified in SAE J1772.

IEEE 1547 identifies three protocols that can be used to meet the requirements and IEEE 2030.5 is one of them. SAE J2847/3 serves as a user guide to the DER function set of IEEE 2030.5. The SunSpec IEEE 2030.5 V2G-AC Profile document goes into further detail to map IEEE 2030.5 function sets, steps of operation, and messages for the EV application of the protocol as shown in Table 14.

NOTE: The purpose of Table 14 is to outline the informative example in the V2G AC Profile and the requirements in the documents need to be considered.

Table 14 - IEEE 2030.5 to SAE J3072 communication steps and signals

Step	Description	Sub #	Action	Signals
0	PEV plugs into EVSE and establishes communications	5.1	Service Discovery	
		5.2	Resource Discovery	PEV gets Device Capability.
				PEV gets Time.
				PEV gets EndDeviceList.
				PEV gets its DERList.
1	PEV obtains site limits information from the EVSE	5.3	PEV Gets Site Limits	PEV gets EVSE's SelfDevice.
				PEV gets the EVSE's DERList.
				PEV gets EVSE's DERSettings.
2	Based on the EVSE site limits, PEV sends back its adjusted configuration settings	5.4	PEV Sends Info to EVSE	PEV put its Device Information.
				PEV puts its PowerStatus which contains the PEVInfo resource.
				PEV puts its DER Capability.
				PEV puts its adjusted settings based on the site limits it got from the EVSE.
				PEV put its DER Availability.
				PEV put its DERStatus.
3	EVSE provides Management Information (e.g., DER curves or controls) that are different than the IEEE 1547 defaults	5.5	PEV Gets Management Information	PEV gets its Function Set Assignments List.
				PEV gets Time. Note: This may be different than the Time in Step 0.
				PEV gets its DER ProgramList.
				PEV gets its DefaultDERControl.
				PEV gets it's DERControlList. Note: This may be three items at a time until complete.
		5.6	PEV Gets Management Information Curves	PEV gets the Frequency Droop curve.
				PEV gets the HFRT Must Trip curve.
				PEV gets the HVRT Momentary Cessation curve.
				PEV gets the HVRT Must Trip curve.
				PEV gets the LFRT Must Trip curve.
				PEV gets the LVRT Momentary Cessation curve.
				PEV gets the LVRT Must Trip curve.
				PEV gets the Volt-Var curve.
				PEV gets the Volt-Watt curve.
		5.7	PEV Responses	PEV posts its DERControl Response for each DERControl - Started for the nine DERControls.
4		5.8	PEV Sets Up Metrology	PEV posts its Mirror Usage Point.
				PEV gets the Mirror Usage Point.
				PEV creates the Active Power MMR.
				PEV creates the Reactive Power MMR.
				PEV creates the Voltage MMR.
				PEV creates the Frequency MMR.
				PEV posts a new Active Power reading.
				PEV posts a new Reactive Power reading.
				PEV posts a new Voltage reading.
				PEV posts a new Frequency reading.
		5.9	Subscriptions and Notifications	PEV creates a subscription to the DERControlList.
				EVSE posts a Notification to the PEV.
				PEV creates a subscription to the Default DERControl.
				EVSE posts a Notification to the PEV.
5	EVSE authorizes PEV to discharge	--	Periodic GETs of DefaultDERControl	PEV gets the Default DERControl

Step	Description	Sub #	Action	Signals
6	PEV continuously verifies that it is still authorized to discharge	5.10	Periodic Gets of Information	PEV gets the Default DERControl
				PEV gets the DERControlList
				PEV monitors its EndDevice instance.
				PEV monitors its MUP instance.
		5.11	PEV Sends Periodic Information	PEV puts its DERStatus
				PEV puts its PowerStatus which contains the PEVInfo resource.
				PEV put its DER Availability.
				PEV posts its Meter Readings - Active Power.
				PEV posts its Meter Readings - Reactive Power.
				PEV posts its Meter Readings - Voltage.
				PEV posts its Meter Readings - Frequency.
		5.12	Other Functions	PEV gets new DERControl.
				PEV posts its DERControl Response - Received.
				PEV posts its DERControl Response - Started
				PEV gets new DERControl.
				PEV posts its DERControl Response - Received.
				PEV posts its DERControl Response - Started
7	Session termination and teardown			

5.2 AC V2G - EVSE Requirements

The AC EVSE is required to meet UL 1741 Supplement C requirements and the evolving market (utility) needs.

The onboard charger (OBC) is not required to include any GFCI since this is included in the EV EVSE.

5.3 Boundary Diagrams

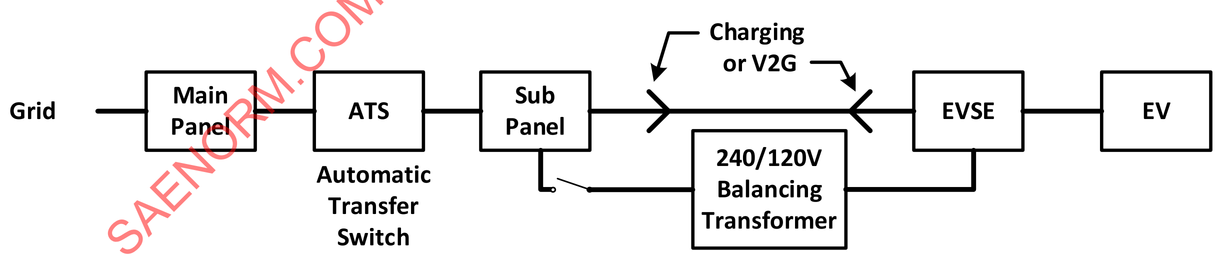


Figure 21 - Charging or V2G and V2H functional diagram

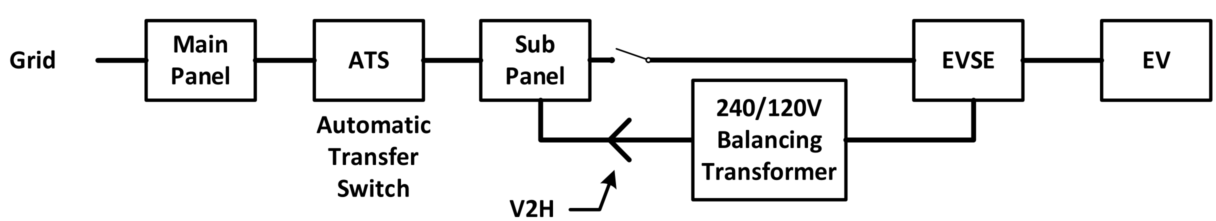


Figure 22 - V2H functional diagram

A detailed system schematic is shown in Figure 23.

An auto transformer is shown; however, an isolation transformer or a neutral driving circuit are other options.

The main breaker is included as the automatic transfer switch, and these could be separate functions.

The manual disconnect is expected to be located outside the home next to the main meter and is required for some utilities.

This schematic is the complete system, and a separate panel is another option to separate critical loads for V2H, if desired.

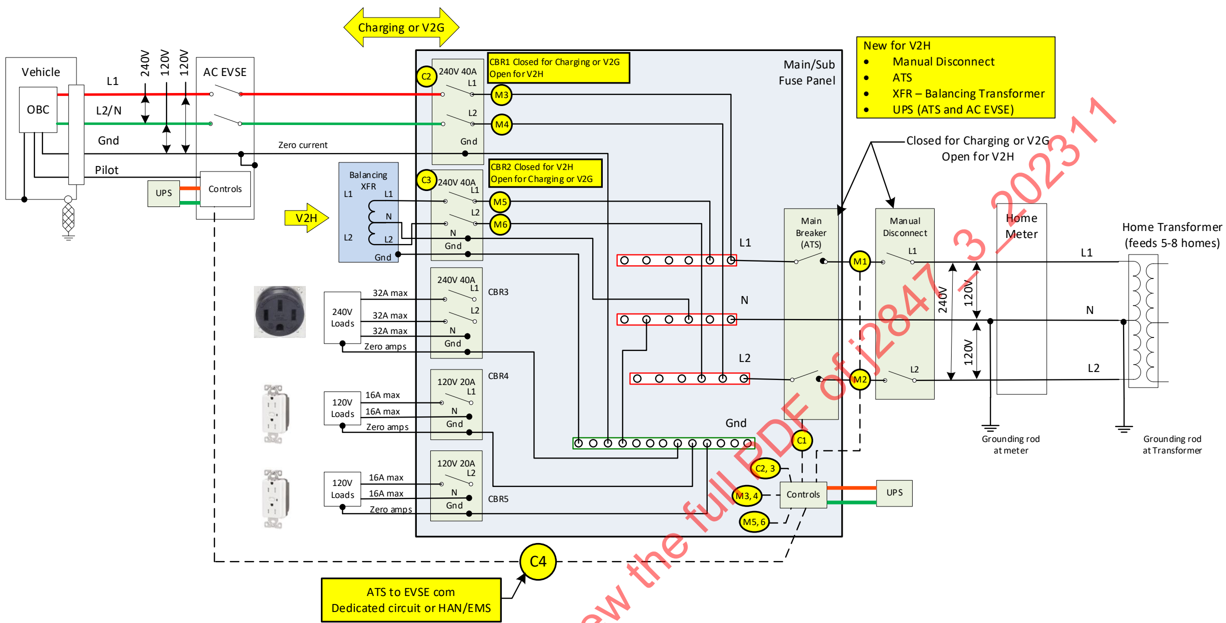


Figure 23 - Detailed wiring diagram

Controls and monitoring summary are shown in Table 15.

Table 15 - V2G and V2H controls and monitoring summary

Grid	Function	Component	Position	Control	Monitor
Yes	Charging or V2G	Main breaker (ATS)	Closed	C1	
		CBR1	Closed	C2	
		CBR2	Open	C3	
Goes out	V2H	Main breaker (ATS)	Open	C1	M1, M2
		CBR1	Open	C2	
		CBR2	Closed	C3	
Returns	Stop V2H	Main breaker (ATS)	Open	C1	M1, M2
		CBR1	Closed	C2	M3, M4
		CBR2	Open	C3	M4, M5
Yes	Restart Charging or V2G	Main breaker (ATS)	Closed	C1	
		CBR1	Closed	C2	
		CBR2	Open	C3	

5.4 Sequence Diagrams and SAE J1772 Pilot Stages

Use cases are shown in Table 16.

Table 16 - V2G use case summary

Initial Request	Authorization	Current Request	Action	Pilot Frequency State B2
V2G Requested	Authorized	V2G Requested	EV opts into grid functions with importing and exporting active power capability	1 kHz
V2G Requested	Authorized	V2G not requested	EV opts out of exporting active power	1 kHz
V2G Requested	Authorized	V2G not requested	EV opts out of grid functions and charges instead	1 kHz
V2G Requested	Not authorized	Does not matter	EV charges	1 kHz
V2G not requested	Does not matter	V2G not requested	EV charges	1 kHz
Does not matter	Does not matter	V2H requested	EV provides energy to home (non-exporting, emergency backup)	166 Hz

The communication for AC V2G is shown in Figure 24. The HAN (not shown) is required if the site power can exceed the inverter and/or the home balancing transformer capabilities.

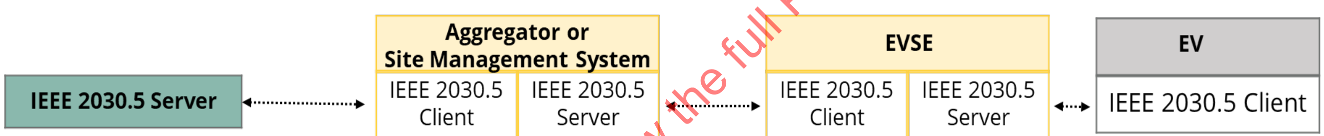


Figure 24 - V2G communication summary

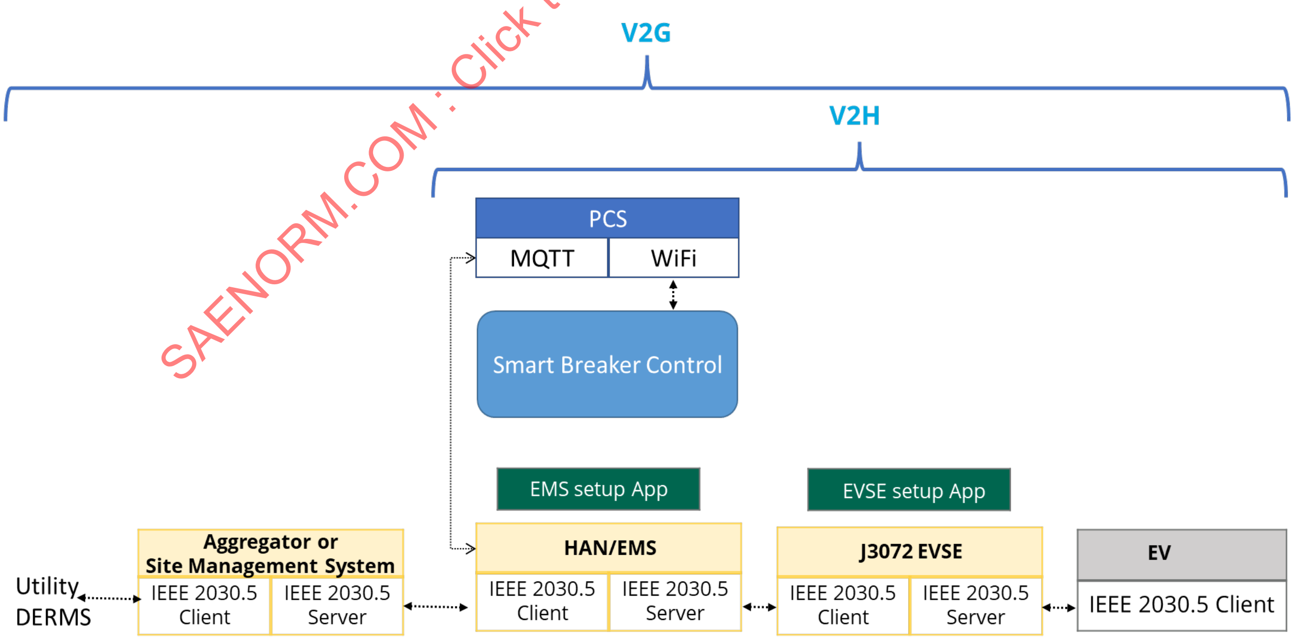


Figure 25 - V2H energy management system summary

Operational Scenarios are shown in Figure 26. Figure 27 shows more detail for opting into the V2G session and Figure 28 shows the EV opting out of the V2G session.

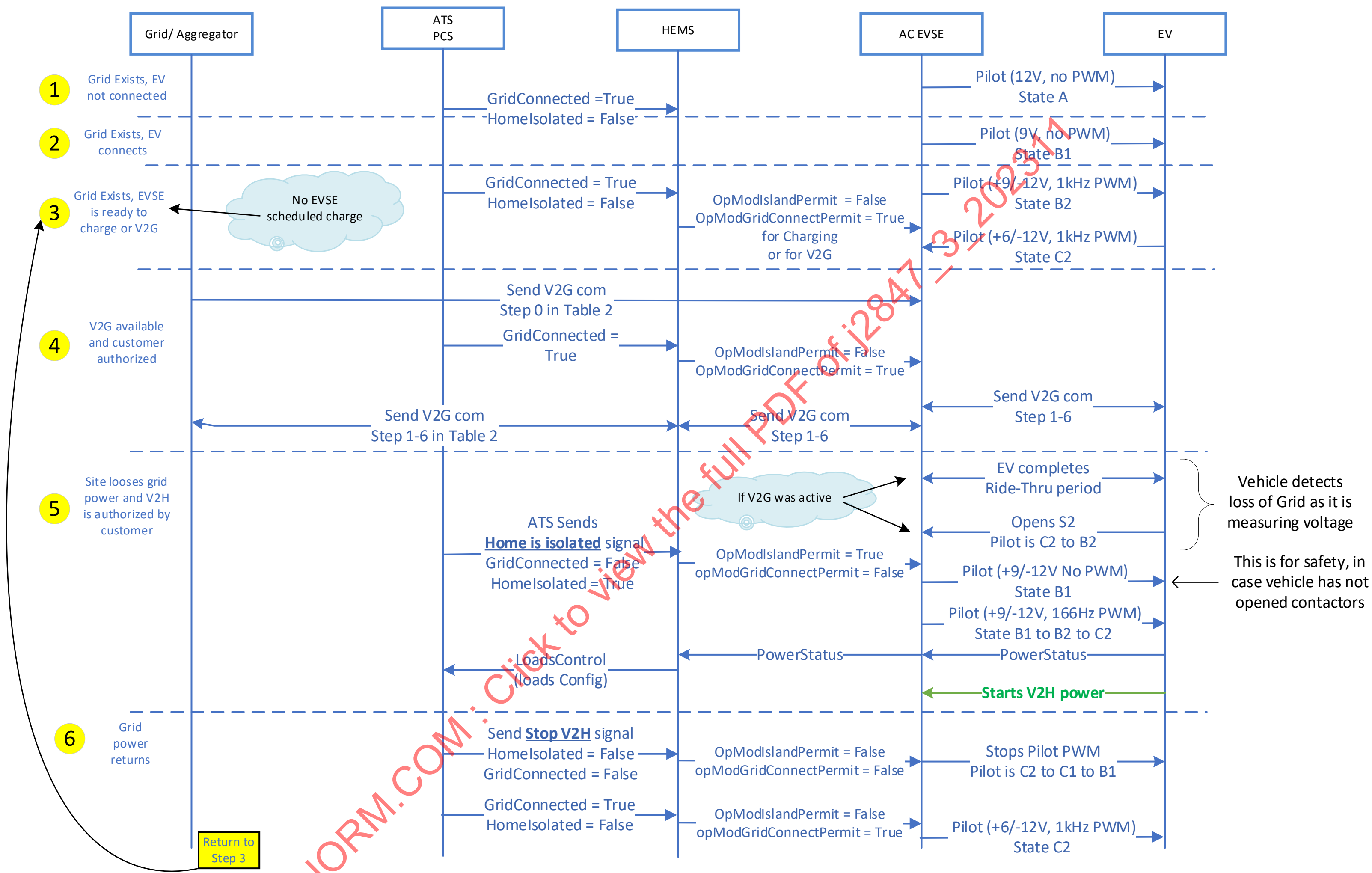


Figure 26 - Operational scenarios

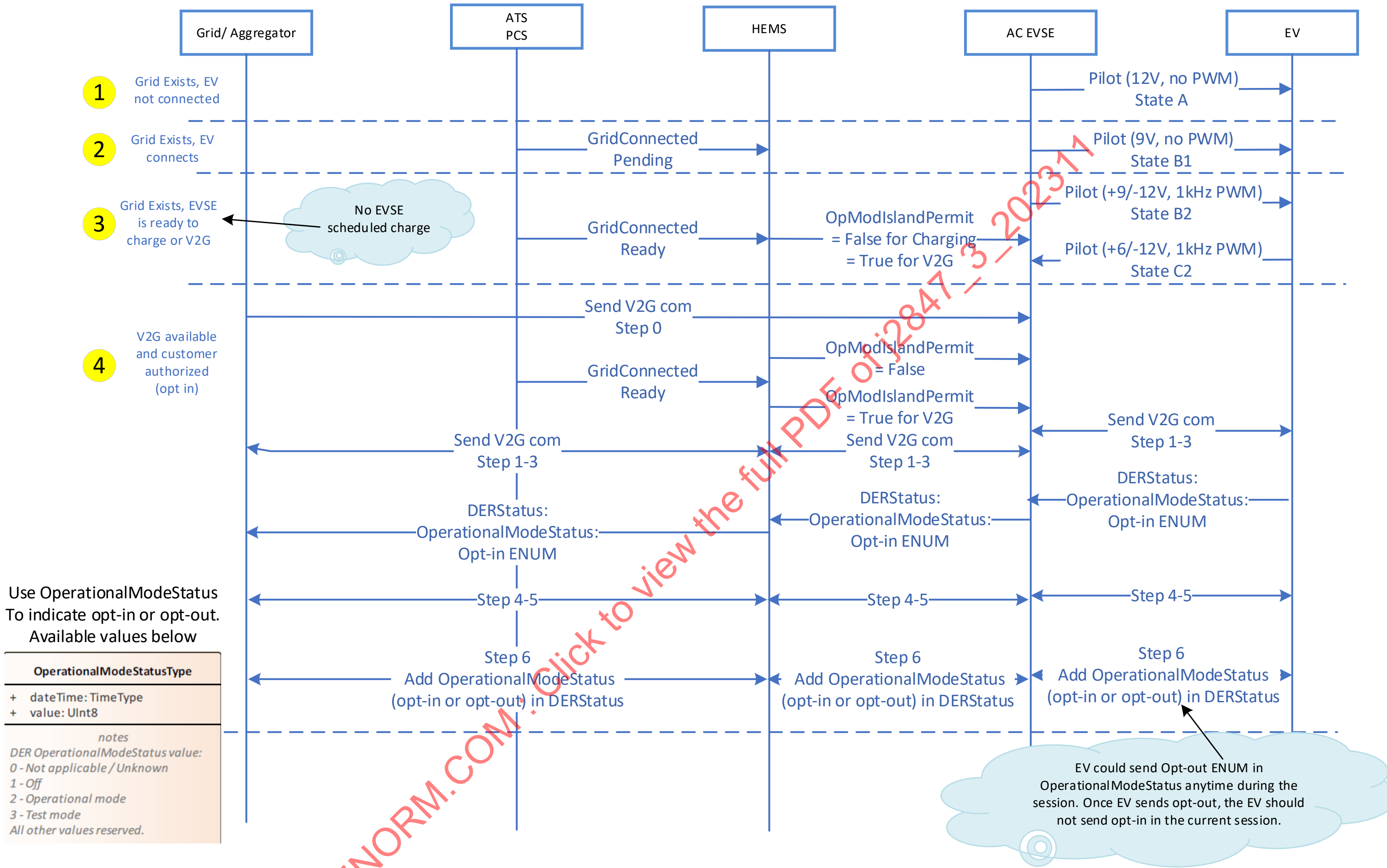


Figure 27 - Opting into V2G session

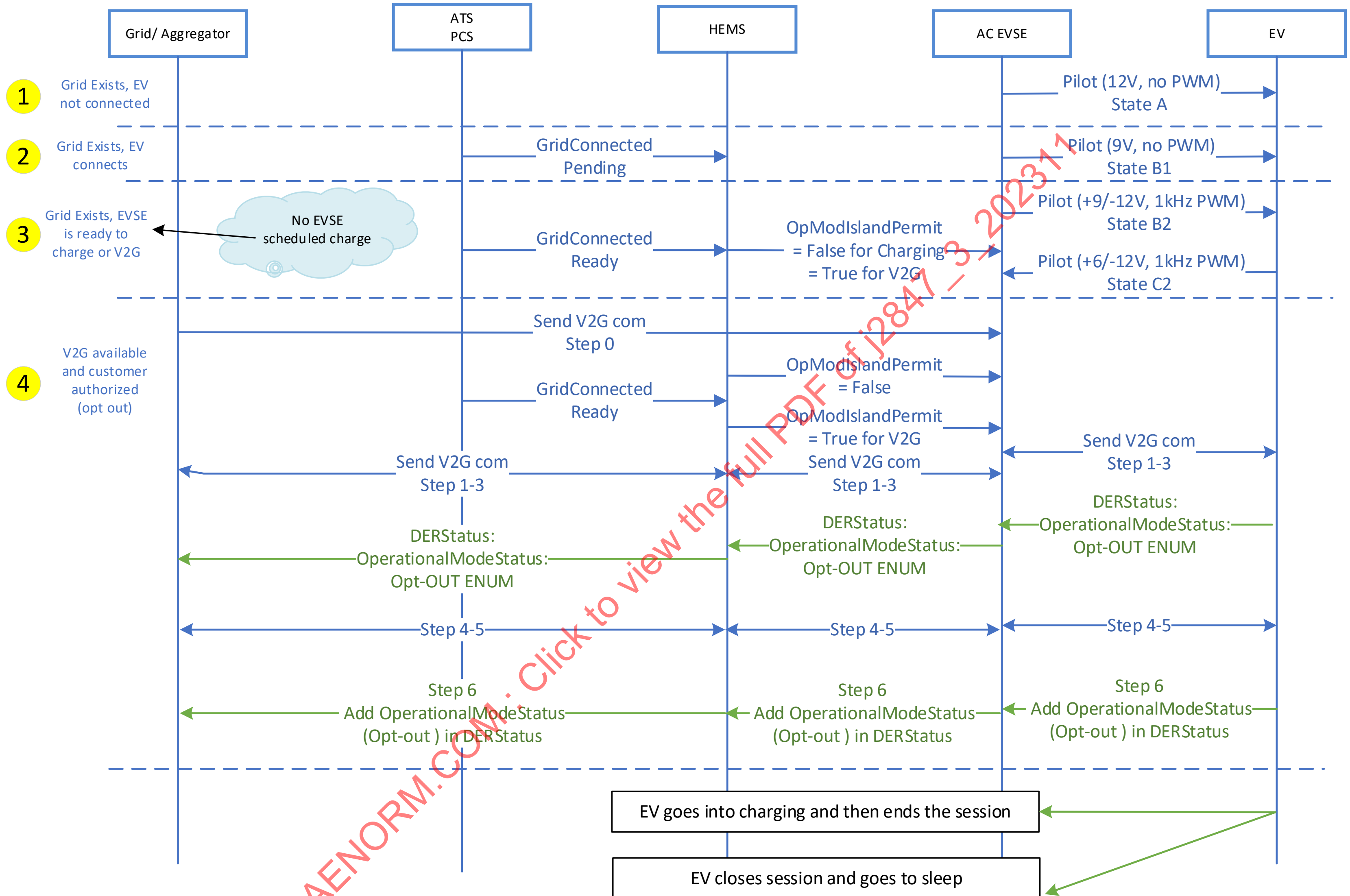


Figure 28 - Opting out of the V2G session

IEEE 2030.5 Commands:

opModIslandPermit attribute (boolean) [0..1]
Permits (true) or disallows (false) grid islanding. This control is likely to be more useful for microgrid controllers.

opModEnergize attribute (boolean) [0..1]
Set DER as energized (true) or de-energized (false). Used in conjunction with ramp rate when re-energizing. If both opModConnect and opModEnergize are present, the values are logically ANDed to determine the connection state.

The valid Pilot states and transitions are shown in Figure 29 from EV connection to energy transfer.

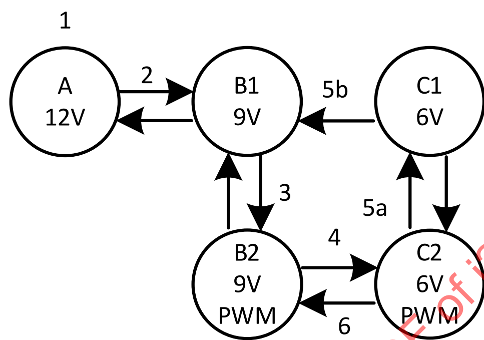


Figure 29 - EVSE-EV valid energy transfer pilot states and transitions

Normal sequence when EV connects and is ready to transfer energy is A to B1 to B2 to C2, shown in Figure 30.

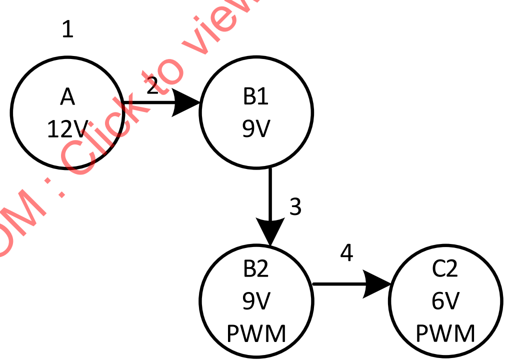


Figure 30 - Normal energy transfer state sequence

Figure 31 identifies the steps for the EVSE to stop the energy transfer session. State C2 to C1 to B1.

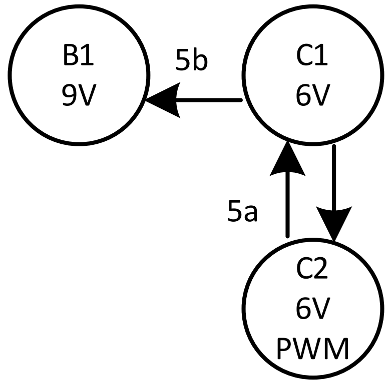


Figure 31 - EVSE stops the energy transfer session

Figure 32 identifies the steps for the EV to stop the energy transfer session. State C2 to B2. This may be used when the OBC detects islanding per the IEEE 1547-2018 Section 8 requirement and stop discharging.

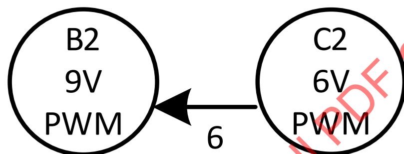


Figure 32 - EV stops the energy transfer session

6. ADDITIONAL AC V2H ITEMS

6.1 System Components

1. AC EVSE may have to be registered with the utility to only operate if the home is islanded from the grid through an automatic transfer switch (disconnect means).

    a. May also be required to be hard wired, not pluggable (also in NEC® Article 702 versus 705).

2. Automatic transfer (islanded, isolated, etc.) switch (ATS) function (grid disconnect).

    a. At the home main power panel. Transfer switch isolates home from grid.

        i. Signals the AC EVSE that the home is islanded and AC V2H can start.

        ii. Monitors grid power returning and notifies home system so V2H session can stop.

        iii. More about the assurance needed.

3. Home balancing transformer.

    a. The EV provides balanced 240 VAC to the home and this transformer balances the 120/240 V loads in the home, same as the grid transformer functions when the grid power is connected.

4. Home area network (HAN) is also required for AC V2G management of home loads.

    a. Provides the home management function to ensure the home loads do not exceed the power capability of the vehicle OBC and/or the balancing transformer.

        i. EV provides FlowReservation or power status info (energy, power, and Time Charge Is Needed) and EV SoC (determines lowest level allowed before stopping energy flow).

    b. Establishes the home load profiles manage with available EV power (each appliance load, cycle time duration, on/off schedule, etc.).

    c. Signals smart breakers when to close and open for the home loads to not exceed EV max power.

        i. May directly interface with home thermostat for heating and cooling control.

5. Cold start battery or UPS for ATS and AC EVSE controls.

6.2 Pilot Signal Communication Summary

The EVSE and vehicle exchange the Pilot signal analogue communications when connected per J1772.

6.2.1 Standby Battery Requirements

The home "cold start battery" or UPS powers the automatic transfer switch controls and the AC EVSE controls. This can be a single or separate units based on energy size requirements and locations of the AC EVSE in the home. This may however only be sufficiently sized to power the EVSE Pilot signal for State A that changes to State B when the vehicle connects. Additional power from the vehicle may be used to power relays in the EVSE and additional power up requirements after the Pilot changes to State C, prior to powering the home.

V2H is always initiated by the EVSE and a Pilot unique frequency of 166 Hz (instead of the 1 kHz) is required to identify to the vehicle that V2H is requested, not a charging or V2G session that uses the 1 kHz frequency.

This frequency variations lets the vehicle bidirectional onboard charger to start as a voltage source or grid forming, instead of the charging mode of constant current, then constant voltage or the V2G mode where the OBC is a current source or grid following.

The Pilot PWM values are the same for V2H as for charging or V2G to indicate available line current (rms) and range from 10 to 96%.

The Pilot frequency is coordinated with other AC V2X functions and other regions as follows.

- Type 1 connector: If 125 Hz Pilot frequency, outputs 120 VAC current corresponding to the PWM value → AC V2L only

- Type 1 connector: If 166 Hz Pilot frequency, outputs 240 VAC current corresponding to the PWM value → AC V2H or V2V.

- GB/T AC connector: If 250 Hz Pilot frequency, outputs 220/380 VAC current corresponding to the PWM value → Single or three phase for AC V2V or V2L.

- Type 2 connector: If 500 Hz Pilot frequency, outputs 230/400 VAC current corresponding to the PWM value → Single or three phase for AC V2V or V2L.

The Pilot frequency tolerances are shown in Table 17 and Figure 33. The frequency spacing is based on even time increments.

Table 17 - Frequency and tolerance summary

Output	Frequency	Time (seconds)				
		Nominal	Max	Min	Range	Gap
120 VAC	125	0.008	0.00870	0.00735	0.00134	
240 VAC	166	0.006	0.00658	0.00546	0.00111	0.00077
220/380 VAC	250	0.004	0.00441	0.00366	0.00074	0.00106
230/400 VAC	500	0.002	0.00221	0.00182	0.00039	0.00145
Charging/V2G	1000	0.001	0.00102	0.00098	0.00004	0.00080

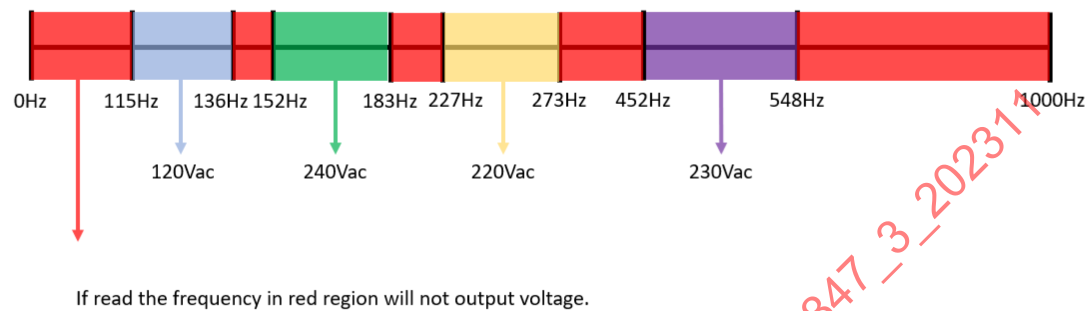


Figure 33 - Frequency tolerance diagram

6.3 ATS to AC EVSE Communication Summary

The ATS shall signal the AC EVSE that the home is islanded and V2H can be initiated. The V2G OBC also detects islanding per IEEE 1547:2018, Section 8 requirement. The EVSE can then initiate the 166Hz Pilot frequency for V2H, after the following communication is complete.

- A communication cable is required between the ATS and AC EVSE and/or Stationary Storage.
- A RS 485 cable can be used to support bidirectional communication between the ATS and AC EVSE

The communication layers presented in Figure 34 and described below are used to implement communication betweem the ATS and AC EVSE over a RS 485 cable.

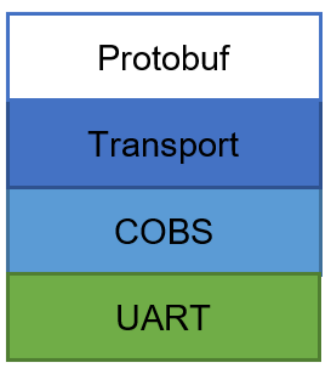


Figure 34 - EVSE-ATS communication layers over UARTs

1. Protobuf: Provides serialization and deserialization of messages.

2. Transport layer: Provides ACK/NAK/replay functionality. Prepends a 1-byte header to each frame, with the following fields:

   a. 2 bits: Protocol version number.

   b. 3 bits: 0 - Request or event (unsolicited); 1 - response; 2 - ack; 3 - nack; 4 - unreliable data (message for which ack/nack and sequence numbering is ignored). A nack indicates the reception of a message known to be bad and triggers a retransmit, starting after the last know good message.

   c. 3 bits: sequence number. Sequence number of the message, Sequence number of the ack'ed message, or sequence number of the last known good message (in the case of a NAK).

4. COBS: Provides message delineation over streams. A trailing CRC-32 is appended to the frame before the cobs encoding step in the Tx direction (and verified in Rx) with the following markers:

   a. 0x00: Start marker

   b. 0xff: 254 bytes jump

4. UART: HW interface for rs-485.

Communication should be implemented using the following interlayer primitives and message primitives

1. Interlayer communication primitives:

   a. Between protobuf and transport layer:

      i. From protobuf to transport layer: "Transmit(msgType, message)"

      ii. From transport layer to protobuf: "Receive(msgType, message)"

   Where msgType is of type of event, request, reply, or unreliable

   b. Between transport layer and COBS:

      i. From transport layer to COBS: "submit frame"

      ii. From COBS to transport layer: "receive frame" without any parameter.

2. Message primitives:

   a. EVSE-ATS messages are blocks of variable-sized, frame-based, binary data. They are encoded in protobuf version 3 format. Note: the following convention allows for the communication of data linked to additionnal/optional features that may be included in some ATS models (e.g., current sensor).

```
syntax = "proto3";
enum AcLineState {
  ON_GRID = 0;
  OFF_GRID = 1;
  READY_FOR_ON_GRID = 2; // Meets all conditions to go ongrid - waiting for a
set_state request
}
enum AcLineBoardType {
  UNKNOWN = 0;
  HOME_CURRENT_SENSOR = 1;
  AUTOMATIC_TRANSFER_SWITCH = 2;
}
```

```
message AcLineGetVersionRequest { }
message AcLineGetVersionResponse {
  int32 majorNo = 1;
  int32 minorNo = 2;
  int32 patchNo = 3;
  int32 buildNo = 4;
  AcLineBoardType board_type = 5;
  bytes revId = 6;
}
message AcLineErrors {
  bool acPresenceError = 1;
  bool acFrequencyError = 2;
  bool acRmsError = 3;
  bool homeRelayError = 4;
  bool j1772RelayError = 5;
}
message AcLineReadings {
  float l1_current = 1;
  float l2_current = 2;
  uint32 ac_voltage = 3;
  float temperature = 4;
  float ac_frequency = 5;
  bool is_ac_present = 6;
  repeated bool relay_closed = 7;
  AcLineErrors errors = 8;
  AcLineState grid_state = 9;
  float phase_offset = 10;
}
message AcLineResponses {
  AcLineGetVersionResponse get_version = 1;
}
message AcLineSetStateRequest {
  AcLineState state = 1;
}
message AcLineSetConfigRequest {
  uint32 on_grid_delay = 1;
}
message AcLineResetRequest { }
message AcLineMessage {
  one of msg {
    AcLineGetVersionRequest get_version = 1;
    AcLineResponses rsp = 2;
    AcLineReadings data = 3;
    AcLineSetStateRequest set_state = 4;
    AcLineSetConfigRequest set_config = 5;
    AcLineResetRequest reset = 6;
  }
}
```

6.4 Required Signals

The ATS should regularly send (without necessarily receiving a request from the EVSE) “AcLineMessage.msg.data” messages. This must be used to allow the EVSE to get the following information as well as to send the following command:

1. Home system is capable of V2H -> if AC EVSE receives messages regularly and without error, it knows that it is connected to a properly functioning ATS. By sending an AcLineGetVersion, the AC EVSE can also confirm that it is connected to a compatible ATS. The field grid_state==ON_GRID, indicates that "grid is powering the home and that the AC EVSE can start an EV charging or V2G session (and that it cannot initiate a V2H session).

2. ATS has islanded the home -> grid_state==OFF_GRID

3. Grid power has returned and V2H session can stop -> grid_state==READY_FOR_ON_GRID

4. EVSE commands to the ATS to reconnect the home to the grid -> AcLineMessage.set_state.state==ON_GRID

6.4.1 Test Signals

Test the system to make sure it functions when needed.

The EVSE and/or ATS can include activation tests using messages defined above to verify that command signals are properly exchanged between EVSE and ATS. This could also be at an automatic cycle without customer intervention. This is to test the home system and may or may not include the EV in this system test.

6.4.2 Additional Signals

Items to consider for additional signals:

1. Customer desires to stop V2H.

    a. Presses S3 on EVSE connector changes from State C to State B, same as stopping charging session.

    b. Presses stop button on EVSE.

    c. Note that ATS continues to monitor grid and home voltage and grid voltage returns, connects to grid as long as home voltage is still zero.

2. Home has ESS and it does grid forming (voltage source), EV then needs to do grid following (current source)

    a. The ATS signal needs to be directed to the unit expected to perform the grid forming function. If this is the ESE, the AC EVSE should also receive a signal to inform it of the house islanding status and the presence of another grid forming inverter to allow it to manage EV charging or discharging functions accordingly. For example, the EV may not be able to charge, depending on home loads and ESS power that requires additional coordination signals.

3. Solar energy is present, and ESS and EV may be charging versus discharging.

7. VEHICLE-TO-GRID USE CASES

Options for communication paths to the vehicle are shown in Figure 35. Figure 36 identifies the additional or other sources and loads to be considered in the home.

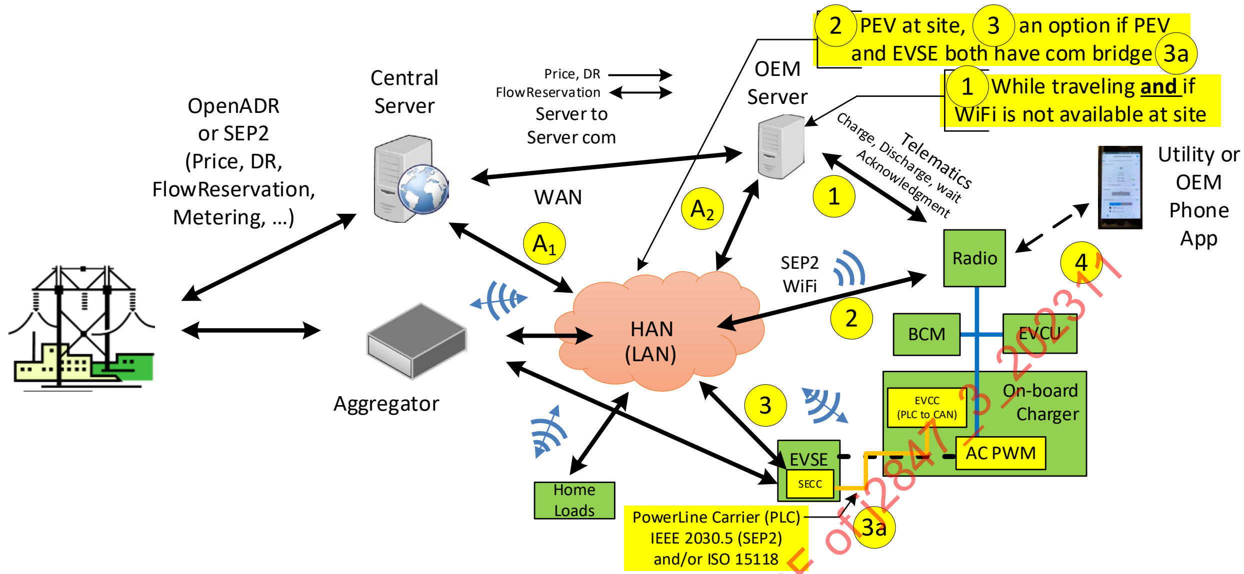


Figure 35 - Communication paths

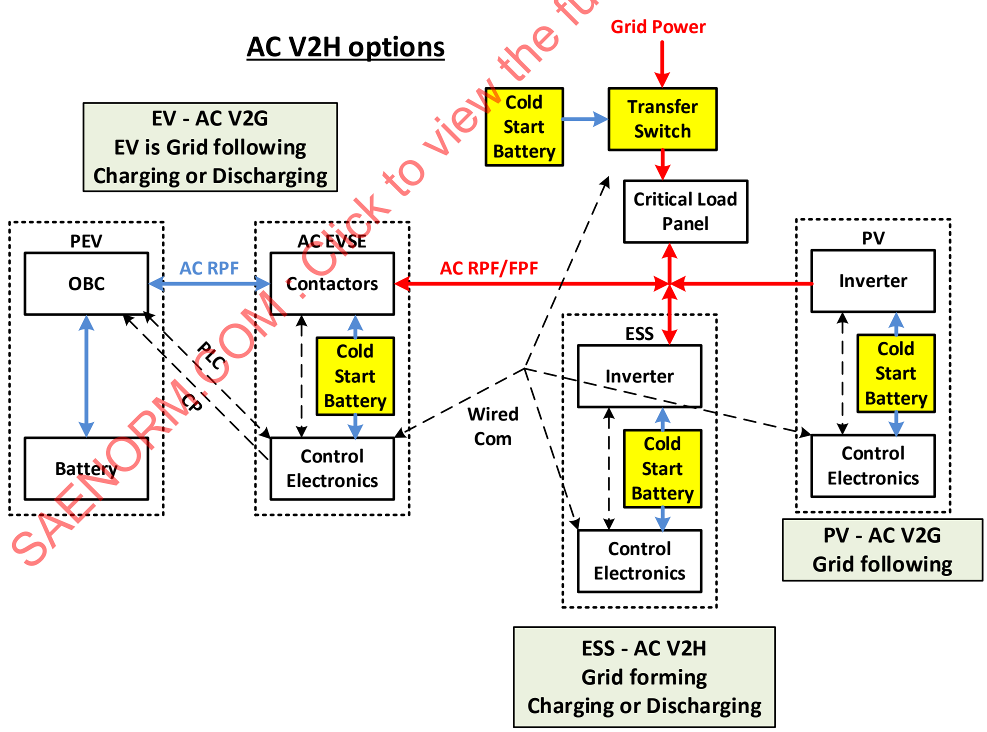


Figure 36 - Additional loads and sources in the home

Use Cases (V2H):

1. EV only:

    a. EV is Grid Forming

2. EV and ESS:

    a. ESS is Grid Forming (if sufficient SoC)

    b. EV is Grid Following

EV and PV:

a. EV is Grid Forming

b. PV can charge EV if excess power is available

EV and ESS and PV:

a. ESS is Grid Forming (if sufficient SoC)

b. EV is Grid Following

c. PV can charge EV and/or ESS if excess power is available

Add info on PV providing energy to charge vehicle during V2H when home loads are less than generated.

8. NOTES

8.1 Revision Indicator

A change bar (l) located in the left margin is for the convenience of the user in locating areas where technical revisions, not editorial changes, have been made to the previous issue of this document. An (R) symbol to the left of the document title indicates a complete revision of the document, including technical revisions. Change bars and (R) are not used in original publications, nor in documents that contain editorial changes only.

PREPARED BY SAE HYBRID - EV COMMITTEE

APPENDIX A - ABBREVIATIONS

A	Amperes
AC	Alternating Current
AGC	Automatic Generation Control
AMI	Advanced Metering Infrastructure
BEV	Battery Electric Vehicle
BMS	Battery Management System
CPP	Critical Peak Pricing
DC	Direct Current
DG	Distributed Generation
DMS	Distribution Management System
DNS	Domain Name System
DNS-SD	Domain Name System - Service Discovery
DRLC	Demand Response Load Control
ECP	Electrical Connection Point
EDEV	End Device
EEI	Edison Electric Institute
EMS	Energy Management System
EPP	Exportable Power Panel
EPRI	Electric Power Research Institute
EPS	Electric Power Systems
ER	Distributed Energy Resources
ESCO	Energy Services Company
ESI	Energy Services Interface
EUMD	End Use Measurement Device
EV	Electric Vehicle
EVSE	Electric Vehicle Supply Equipment
EVSP	Electric Vehicle Services Provider
EXI	Efficient XML Interchange
FPF	Forward Power Flow

HAN	Home Area Network
HMI	Human Machine Interface
HTTP	Hypertext Transfer Protocol
HVRT	High Voltage Ride Through
IEC	International Electrotechnical Commission
IP	Internet Protocol
KW	Kilowatts
KWH	Kilowatt-Hours
LAN	Local Area Network
LVRT	Low Voltage Ride Through
MRD	Market Requirements Document
NEC®	National Electrical Code®
NEMA	National Electrical Manufacturers Association
NIST	National Institute of Standards and Technology
OBW	Operational Bandwidth
PAP	Priority Action Plan
PCC	Point of Common Coupling
PEV	Plug-in Electric Vehicle
PLC	Power Line Communication
PTC	Power Transfer Capability
PV	Photovoltaic (solar)
PWM	Pulse Width Modulation
REST	Representational State Transfer
RPF	Reverse Power Flow
RTD	Real-Time Dispatch
SCADA	Substation Control and Data Acquisition
SDEV	Self Device
SEP2	IEEE 2030.5-2018 - Standard for Smart Energy Profile Application
SGIP	Smart Grid Interoperability Panel
SOC	State of Charge

SRS	System Requirement Specification
TCIN	Time Charge Is Needed
TCP	Transmission Control Protocol
TOU	Time of Use
TRD	Technical Requirements Document
UCAIug	UCA® International Users Group
UCAP	Usable Capacity of battery
UDP	User Datagram Protocol
UF	Utilization Factor
UML	Unified Modeling Language
UOM	Unit of Measure
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name, Uniform Random Number
UTC	Universal Time, Coordinated
V	Volts
V2G	Vehicle-to-Grid
V2H	Vehicle-to-Home
VA	Volt-Amperes
VAr	Volt-Amperes Reactive
VSS	Voltage Support Service
W	Watts
WADL	Web Application Description Language
WAN	Wide Area Network
WLAN	Wireless Local Area Network
XFMR	Transformer
XML	Extensible Markup Language
XSD	XML Schema Definition

APPENDIX B - HOME ENERGY MANAGEMENT USING FLOW RESERVATION

A homeowner plans to use the available energy in a PEV to offset residential loads during a period of higher energy prices in the evening and then recharge the PEV battery at lower rates after midnight. A home energy management system will be used to control the rate of discharging of the PEV to offset other residential loads and maintain the total premises demand below a target level. While the home EMS and PEV could perform this application using the fixed flow mode of the DER function set, this example is based on using the flow reservation function set.

B.1 DETAILED DESCRIPTION

A PEV arrives at home shortly before 5:00 p.m. and the driver enters data using vehicle's controls and displays to set up the PEV for a discharging and charging session. It is November 8, 2013, in the Eastern Time zone. The SOC is currently 65%. The driver enters a Time Charge Is Needed of 8:00 a.m. the next day and uses the default Target State of Charge of 100%. The driver authorizes the PEV to engage with the home energy management system (EMS) using the flow reservation function in a program in which the EMS will control the rate of charging and discharging to both limit the total home demand and also ensure that the PEV achieves its target SOC by 8:00 a.m. As part of setting up for a discharge flow reservation the driver selects a start time of 5:00 p.m., duration of 7 hours (or end time of 12:00 a.m.), and a default maximum discharge rate setting of 100% of the maximum available power. The driver also uses a default setting for the minimum SOC for use in DER discharging of 20%. The duration (or end time) could be automatically limited by the latest time that the PEV could discharge to the minimum SOC and still charge from 20 to 100% by 8:00 a.m.

The EMS program is set up to manage the total home demand below 1500 Watts before 12:00 a.m. and to below 5000 W from 12:00 a.m. until 8:00 a.m. The 1500 target was established by the homeowner. The 5000 W target was established by a written request from the utility as part of a preferred PEV rate program. Both values are soft targets and may be exceeded if the EMS cannot offset or shift loads and still meet the PEV charging requirements. Before midnight, the EMS will first seek to manage residential loads to reduce peaks and then call on any available reverse power flow from the PEV to further offset the load to stay below the 1500 W target. The energy prices are most favorable after midnight and the EMS would not schedule the PEV to start charging until that time as long as it is possible to complete charging by 8:00 a.m.

The EMS and PEV will use the flow reservation function for the session. The PEV will automatically make two sequential reservations. The PEV software will calculate much of the information needed for each of the two reservations requests. The first flow reservation request will be to discharge the battery to the minimum SOC at up to the maximum discharge rate. The start time will be 5:00 p.m. and the duration will be until midnight. In this example, it is possible to recharge from 20 to 100% even when starting after midnight. The PEV will cancel the first request at the earlier of whenever the minimum SOC is reached or at midnight and then provide the EMS with a new request for charging. This would be to achieve a 100% SOC by 8:00 a.m. using up to the maximum charge rate. Charging requests must always use the maximum charging rate and the PEV should be not provide the driver with a selection option to use a reduced rate. For both flow reservation requests, the PEV computes the value of the energy request.

For this application example, the EMS will issue a new flow reservation response for the active request every 5 minutes. It is possible that demand could exceed the limit at some point during the 5 minute interval, but since these are economic targets, the average over the hour will be controlled. If the premises had a 15 minute monthly peak demand charge, a 5 minute update may be too coarse and a 1 minute refresh may be needed—but in this example, it is assumed that there is only an energy rate. It may be possible for the PEV to subscribe to the EMS flow reservation Response List and receive the updates directly without polling, but for this example polling by the PEV will be used.

If the EMS is refreshing the response every 5 minutes, it would be unacceptable for the PEV to poll the EMS every 5 minutes and miss an EMS update by only milliseconds. This could cause 5 minute latency. For this application, it should be assumed that latency should be not more than 1 minute. This will require the PEV to poll the EMS every minute for a new response. This means four of every five GETs will not find a new EMS response.

Unlike the DER function, it is not possible when using flow reservation to switch between charging and discharging for a single request. From 5:00 p.m. until midnight, the EMS can only adjust the rate of PEV discharging. After midnight, the EMS can only adjust the rate of PEV charging. The EMS cannot use a very short discharge to clip a peak above 5000 W using flow reservation, but using the DER fixed flow function it could.

A simulation was conducted for a representative load profile for a home without the PEV charging load. This is shown in Figure B1 in blue. The red line shows the power profile when the PEV is connected and being controlled by the EMS. The EMS is able to use a discharging flow reservation request to maintain the 1500 W limit before midnight and a charging flow reservation request to maintain the 5000 W limit after midnight. The PEV successfully completes charging. The total energy demand (area under respective curves) is higher with the PEV because of the added energy demand for charging. However, if unrestricted charging at 6500 W began at midnight without using the EMS, this charging profile could have added to the blue line after midnight and peak power at 1:10 a.m. could have been 11700 W. This simulation was used to create the data used for the selected exchanges used in the example.

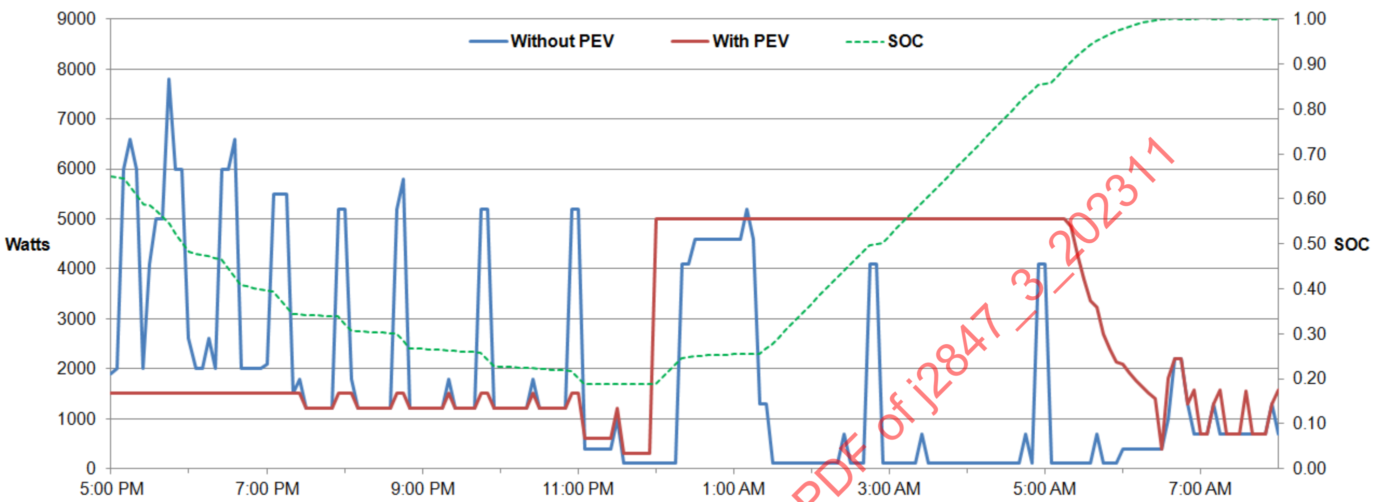


Figure B1 - Home energy management simulation

B.2 SYSTEM ARCHITECTURE

The system architecture to be used for this example is shown in Figure B2. The EVSE is set up to directly bridge the PEV with the HAN. The PEV is a client device and hosts its end-device resources on the EMS server. The EMS uses SEP2, SEP1.x, Matter1.x to communicate with the PEV and all other smart energy devices on the HAN. The EMS may also access resources on the internet as needed. In this example, only the communications between the PEV and the EMS will be shown. The EMS communications with the EVSE, meter, home appliances, and the internet are beyond the scope of this example.

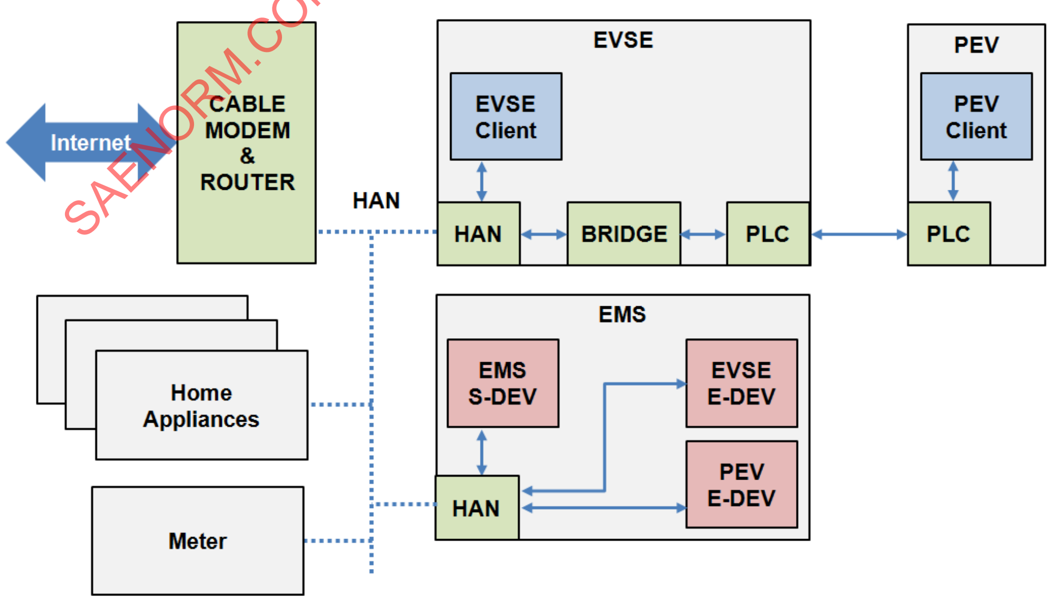


Figure B2 - System architecture for home energy management example

B.3 EMS AND PEV DESIGN CONSIDERATIONS

SEP2 deals with the communication between the PEV and the EMS. It does not define the controls and displays needed in the PEV and EMS to initially setup the application or to program the PEV and EMS for a specific session. The SEP2 standard does provide some rules of engagement, such as event conflict resolution, but these are guidance for developing application software for the devices and are not part of the actual communications package.

The PEV must have controls and displays and associated software to allow the flow reservation function to be used for this application example. The generic structure would be established by the vehicle manufacturer. The PEV may provide certain data entry options and selections to be made during the initial setup for the specific enrolled application. Other data entry would be made after the PEV is connected for a specific session. This would be not unlike working with the various selections on a vehicle navigation and media console.

When a PEV enrolls in an application, even with a home EMS, setup instructions would be provided. For example, the polling rate for flow reservation responses would need to be entered. The refresh rate needed by the EMS for the power status Resource would be entered. Note that with the 2018 update of SEP2, the EMS can inform the PEV of the desired polling rate using the appropriate *pollRate* attribute. In addition, the EMC can inform the PEV of the desired refresh rate using the appropriate *postRate* attribute. The minimum SOC for discharging could be defined for this application. Resource addresses, security settings, and other parameters needed for any use by this application would be provided. For the example, the PEV is setup to poll for a new response every minute at the exact minute mark. The PEV is setup to refresh power status on the end-device server every 5 minutes at the exact 5 minute mark.

For a specific session the driver should always enter the Time Charge Is Needed and the Target SOC, if the default of 100% is not used. The driver would enable discharging, which may have some additional selections, such as entering a minimum SOC or using the application default value. Then the driver would select and authorize the specific application which is based on using a discharge flow reservation. The PEV would prompt the driver to enter or select the earliest start time for discharging, maximum duration (or end time) for discharging, and a maximum authorized discharge rate (as either a percent of the maximum available or as an explicit value in Watts). The PEV software will be able to calculate all of the remaining values needed for both the discharging and charging requests needed by the home EMS to perform the application.

This application example is based on an EMS program that uses the SEP2 flow reservation function to engage with a PEV. The EMS will maintain no more than one request from the PEV. The PEV must cancel an active request and provide a new request to change over from discharging to charging. The EMS is programmed to update the responses every 5 minutes at exactly 10 seconds before the start of each 5 minute mark. The associated event start time would be exactly on the 5 minute marks of each hour. The EMS will be designed to allow the target home demand to be set to different values throughout the day. Interaction of the EMS with other devices in the home and its energy management algorithms are beyond the scope of the example.

B.4 TIMELINE FOR SELECTED EXCHANGES

This is a partial timeline for selected exchanges related to the use of theSEP2 flow reservation function for this application example. The focus here will be on defining the information content used in the XML payloads that will appear in the step-by-step Table B1.

11/8/2013 16:50 EST (1383947400) - Setup and Initialization

The driver performs the following data entry and selection using PEV controls and displays:

- Time Charge Is Needed: 11/9/2013 at 8:00 a.m. EST (1384002000)
- Target State of Charge: Uses default of 100%
- Selects the enrolled Home Energy Management application (this would require initial setup using PEV software to define that it is based on a single discharge flow reservation followed by an automatic PEV generated charge flow reservation)
- Minimum Discharge SOC: Uses default of 20%

- Earliest time to start discharging: 11/8/2013 at 5:00 p.m. EST (1383948000)
- Latest time to stop discharging: 11/9/2013 at 12:00 a.m. EST (1383973200)
- Maximum Discharge Rate Request: Uses default of 100% of maximum discharge rate

Vehicle State at Connection (known or computed by vehicle software):

- Effective maximum charge and discharge rate: 6500 W (nameplate 6600 W)
- Usable battery capacity: 23760 W·h (nameplate 25000 W·h)
- Parasitic losses: 200 W
- State of Charge at connection: 65%
- Charger/Inverter efficiency: 90%
- EVSE Control Pilot: 32 A (7.7 kW at 240 VAC)

The PEV initiates communication when it is connected to the EVSE and the HAN. The PEV would perform registration, certificate exchanges, discovery, function set assignments, time synchronization, and other processes needed at the beginning of any session. Most of these exchanges will be generic for any event-based application and will not be discussed in this document or included in the step-by-step descriptions.

11/8/2013 16:55 EST (1383947700) - Request to Discharge

PEV computes values for the discharging reservation request and posts it to EMS. The time of the first flow reservation request is assumed to be just after the PEV has been connected to the EVSE, the driver data entry has been completed, and the housekeeping communications have been completed.

- durationRequested: 5330 (calculated by PEV as 3600 x 9623 / 6500)
- energyRequested = -9623 (calculated by PEV as 23760 x (0.65-0.20) x 0.9)
- intervalRequested.Start: 1383948000 (based on driver entry of 11/8/2013 5:00 p.m.)
- intervalRequested.Duration: 25200 (7.0 hours calculated using driver entry of latest stop at midnight)
- powerRequested: 6500 (calculated as lower of EVSE pilot and effective inverter capability for 100% selection)

Home EMS will acknowledge request was received but will not immediately provide a response. In this example, the PEV software will assume that a response will not be posted by the EMS earlier than the requested start time or that even if one is posted the PEV will not need to poll for the first response until the requested start time. An earlier start time should never be authorized for a discharge flow reservation request because of safety considerations. If the PEV does not find a valid response within 5 minutes of the requested start time, it should terminate the flow reservation request.

11/8/2013 16:59:50 EST (1383947990) - First Response Posted by EMS

The home EMS computes values for the first response to the discharge request and it is posted to the flow reservation response list server.

- energyAvailable: -9623 (set to match request and never changed in updates)
- powerAvailable: 400 (calculated based on EMS algorithms)
- Event.Interval.Start: 1383948000 (5:00 p.m.)
- Event.Interval.Duration: 600 (use 10 minutes as timeout for all responses)

11/8/2013 17:00 EST (1383948000) - First Response Retrieved by PEV

PEV polls the flow reservation response list server (on the home EMS) for responses. Only one is found and it is executed. The PEV provides a response, if requested. The PEV calculates and posts power status:

- batteryStatus: 4 (mains powered)
- changedTime: 1383948000
- currentPowerSource: 1 (mains)
- estimatedCharge Remaining: 65% (6500)
- chargingPowerNow: -400 W (discharging at EMS command rate)
- energyRequestNow: 10240 W·h
- maxFowardPower: 6500 W
- minimumChargingDuration: 6871 seconds
- targetStateOfCharge: 100% (10000)
- timeChargeIsNeeded: 1384002000 (8:00 a.m. EST 11/9/2013)
- timeChargingStatusPEV: 1383948000

11/8/2013 17:01 EST (1383948060) - PEV Polls Each Minute

The PEV polls flow reservation response list for a response and this is repeated every minute at 17:02, 17:03, and 17:04. No new response is posted by EMS yet.

11/8/2013 17:04:50 EST (1383948290) - Second Response Posted by EMS

The home EMS computes values for the second response to the discharge request and it is posted to the flow reservation response list server.

- energyAvailable: -9623 (set to match request and never changed in updates)
- powerAvailable: 500 (calculated based on EMS algorithms)
- Event.Interval.Start: 1383948300 (5:05 p.m.)
- Event.Interval.Duration: 600 (use 10 minutes as timeout for all responses)

11/8/2013 17:05 EST (1383948300) - Second Response Retrieved by PEV

PEV polls the flow reservation response List server (on the home EMS) for responses. Two are found. The second event is marked as scheduled, it overlaps the first active event, and its start time has been reached or is past. The PEV begins execution of the second response event. The PEV provides a response, if requested. The PEV calculates and posts power status:

- batteryStatus: 4 (mains powered)
- changedTime: 1383948300
- currentPowerSource: 1 (mains)
- estimatedCharge Remaining: 64.77% (6477)
- chargingPowerNow: -500 W (discharging at EMS command rate)
- energyRequestNow: 10302 W·h
- maxFowardPower: 6500 W
- minimumChargingDuration: 6906 seconds
- targetStateOfCharge: 100% (10000)
- timeChargeIsNeeded: 1384002000 (8:00 a.m. EST 11/9/2013)
- timeChargingStatusPEV: 1383948300

11/8/2013 17:06 EST (1383948360) - PEV Polls Each Minute

The PEV polls flow reservation response list for a response and this is repeated every minute at 17:07, 17:07, and 17:09. No new response is posted by EMS yet.

11/8/2013 17:09:50 EST (1383948590) - Third Response Posted by EMS

The EMS computes values for third response to PEV discharge request and posts it. The first response is deleted. Only two are maintained—the active event and the scheduled event.

- energyAvailable: -9623 (set to match request and never changed in updates)
- powerAvailable: 4500 (calculated based on EMS algorithms)
- Event.Interval.Start: 1383948600 (5:10 p.m.)
- Event.Interval.Duration: 600 (use 10 minutes as timeout for all responses)

11/8/2013 17:10 EST (1383948600) - Third Response Retrieved by PEV

The PEV polls the flow reservation response list server (on the home EMS) for responses. Two are found. The first response has been deleted. The list only now has two responses: the second (active) and third (scheduled) event. The PEV begins execution of the scheduled event because its start time has been reached. The PEV provides a response, if requested. The PEV calculates and posts power status (NOT SHOWN IN STEP-BY-STEP EXAMPLE).

11/8/2013 17:11 through 23:05 EST - More of the Same

NO EXCHANGES SHOWN—SIMILAR PATERN REPEATED.

11/8/2013 23:05 EST (1383969900) - Cancelation of Discharge Request

Vehicle Cancels first flow reservation when 20% SOC is reached. The PEV software must retain all of the parameters from the first request because SEP2 cancellation requires the entire content of the original request to be provided.

11/8/2013 23:05 EST (1383969900) - Request to Charge

The PEV computes values for the charging reservation request and posts it to EMS.

- durationRequested: 13641 (calculated by PEV as 1200 + 3600 x 22462 / 6500)
- energyRequested: 22462 (calculated by PEV as (5 x 200) + 23760 x (1.00 - 0.187) / 0.9)
- intervalRequested.Start: 1383973200 (midnight based on TOU schedule)
- intervalRequested.Duration: 28800 (8 hours based on 12:00 to 8:00)
- powerRequested: 6500 (calculated as lower of EVSE pilot and effective charger capability

Home EMS will acknowledge request was received but will not immediately provide a response. In this example, the PEV software will assume that a response will not be posted by the EMS earlier than the requested start time or that even if one is posted the PEV will not need to poll for the first response until the requested start time. If the PEV does not find a valid response within 5 minutes of the requested start time, it should terminate the flow reservation request.

11/8/2013 23:59:50 EST (1383973190) - First Response to Charging Request Posted by EMS

The home EMS computes values for the first response to the discharge request and it is posted to the flow reservation response list server.

- energyAvailable: -9623 (set to match request and never changed in updates)
- powerAvailable: 4900 (calculated based on EMS algorithms)
- Event.Interval.Start: 1383973200 (12:00 a.m.)
- Event.Interval.Duration: 600 (use 10 minutes as timeout for all responses)

11/9/2013 00:00 EST (1383973200) - First Response to Charging Request Retrieved by PEV

The PEV polls the flow reservation response list server (on the home EMS) for responses. Only one is found and it is executed. The PEV provides a response, if requested. The PEV calculates and posts power status:

- batteryStatus: 4 (mains powered)
- changedTime: 1383973200
- currentPowerSource: 1 (mains)
- estimatedCharge Remaining: 18.70% (1870)
- chargingPowerNow: 4900 W (discharging at EMS command rate)
- energyRequestNow: 22462 W·h

- maxFowardPower: 6500 W
- minimumChargingDuration: 13641 seconds
- targetStateOfCharge: 100% (10000)
- timeChargeIsNeeded: 1384002000 (8:00 a.m. EST 11/9/2013)
- timeChargingStatusPEV: 1383973200

Table B1 - Step-by-step for home energy management example

Step	Description
1	**11/8/2013 16:50 EST (1383947400) - Setup and Initialization** Driver provides all required data entry and connects the PEV to the EVSE. The PEV conducts all of the necessary communications with the home EMS and other devices as needed to prepare for the session. These exchanges will not be shown in this table.
2	**11/8/2013 16:55 EST (1383947700) - Request to Discharge** PEV POSTs a flow reservation request to the Home EMS. POST /edev/3/frq HTTP/1.1 Host: {hostname} <FlowReservationRequest xmlns="urn:ieee:std:2030.5:ns"> <mRID>68512866203db3b10000e566</mRID> <description>Discharge between 5 PM and 12 AM</description> <!-- 11/8/2013 4:55 PM EST --> <creationTime>1383947700</creationTime> <durationRequested>5330</durationRequested> <energyRequested> <multiplier>0</multiplier> <value>-9623</value> </energyRequested> <intervalRequested> <duration>25300</duration> <!-- 11/8/2013 5:00 PM EST --> <start>1383948000</start> </intervalRequested> <powerRequested> <multiplier>0</multiplier> <value>6500</value> </powerRequested> <RequestStatus> <dateTime>1383947700</dateTime> <!-- Requested --> <requestStatus>0</requestStatus> </RequestStatus> </FlowReservationRequest>
3	Home EMS sends the following response. HTTP/1.1 201 Created Location: /edev/3/frq/1
4	**11/8/2013 16:59:50 EST (1383947990) - First Response Posted by EMS** Home EMS computes and posts a flow reservation response to server.

<table>
<tr><th>Step</th><th>Description</th></tr>
<tr><td>5</td><td>11/8/2013 17:00 EST (1383948000) - First Response Retrieved by PEV

GET /edev/3/frp HTTP/1.1
Host: {hostname}</td></tr>
<tr><td>6</td><td>Home EMS sends the following response and the PEV starts to execute the event.

HTTP/1.1 200 OK
Content-Type: application/sep+xml

<FlowReservationResponseList xmlns="urn:ieee:std:2030.5:ns" all="1" href="/edev/3/frp" results="1" subscribable="1">
  <FlowReservationResponse href="/edev/3/frp/1"subscribable="1">
    <mRID>f8afa6fde40db98d0000ea75</mRID>
    <description>Discharge between 5PM and 12AM</description>
    <!-- 11/8/2013 4:59:50 PM EST -->
    <creationTime>1383947990</creationTime>
    <EventStatus>
      <!-- Active-->
      <currentStatus>1</currentStatus>
      <dateTime>1383948000</dateTime>
      <potentiallySuperseded>false</potentiallySuperseded>
    </EventStatus>
    <interval>
      <duration>600</duration>
      <!-- 11/8/2013 5:00 PM EST -->
      <start>1383948000</start>
    </interval>
    <energyAvailable>
      <multiplier>0</multiplier>
      <value>-9623</value>
    </energyAvailable>
    <powerAvailable>
      <multiplier>0</multiplier>
      <value>400</value>
    </powerAvailable>
    <subject>685128662 03db3b10000e566</subject>
  </FlowReservationResponse>
</FlowReservationResponseList></td></tr>
</table>

Step	Description
7	PEV updates Power Status resource on end device server (EMS). PUT /edev/3/ps HTTP/1.1 Host: {hostname} <PowerStatus xmlns="urn:ieee:std:2030.5:ns"> <batteryStatus>4</batteryStatus> <!-- 11/8/2013 5:00 PM EDT --> <changedTime>1383948000</changedTime> <!-- mains --> <currentPowerSource>1</currentPowerSource> <estimatedChargeRemaining>6500</estimatedChargeRemaining> <PEVInfo> <chargingPowerNow> <multiplier>0</multiplier> <value>-400</value> </chargingPowerNow> <energyRequestNow> <multiplier>0</multiplier> <value>10240</value> </energyRequestNow> <maxForwardPower> <multiplier>0</multiplier> <value>6500</value> </maxForwardPower> <minimumChargingDuration>6871</minimumChargingDuration> <targetStateOfCharge>10000</targetStateOfCharge> <!-- 9/23/2013 8:00 AM EDT --> <timeChargeIsNeeded>138400200</timeChargeIsNeeded> <timeChargingStatusPEV>138394800</timeChargingStatusPEV> </PEVInfo> </PowerStatus>
8	Flow reservation server responds. HTTP/1.1 204 No Content
9	**11/8/2013 17:01 EST (1383948060) - Poll for Response** PEV polls for new flow reservation response every minute to avoid data latency. Four of every five will not have any changes. This GET request is for more than a single response if more than one exists. GET /edev/3/frp?l=2 HTTP/1.1 Host: {hostname}

Step	Description
10	EMS responds with same response as earlier and it has already been acted on by PEV. HTTP/1.1 200 OK Content-Type: application/sep+xml <FlowReservationResponseList xmlns="urn:ieee:std:2030.5:ns" all="1" href="/edev/3/frp" results="1" subscribable="1"> <FlowReservationResponse href="/edev/3/frp/1" subscribable="1"> <mRID>f8afa6fde40db98d0000ea75</mRID> <description>Discharge between 5PM and 12AM</description> <!-- 11/8/2013 4:59:50 PM EST --> <creationTime>1383947990</creationTime> <EventStatus> <!-- Active--> <currentStatus>1</currentStatus> <dateTime>1383948000</dateTime> <potentiallySuperseded>false</potentiallySuperseded> </EventStatus> <interval> <duration>600</duration> <!-- 11/8/2013 5:00 PM EST --> <start>1383948000</start> </interval> <energyAvailable> <multiplier>0</multiplier> <value>-9623</value> </energyAvailable> <powerAvailable> <multiplier>0</multiplier> <value>400</value> </powerAvailable> <subject>68512866203db3b10000e566</subject> </FlowReservationResponse> </FlowReservationResponseList>
11	Steps 9 and 10 repeated at 17:02, 17:03, and 17:04.
12	**11/8/2013 17:04:50 EST (1383948290) - Second Response Posted by EMS** Home EMS computes and posts next flow reservation response to server.
13	**11/8/2013 17:05 EST (1383948300) - Second Response Retrieved by PEV** GET /edev/3/frp?l=2 HTTP/1.1 Host: {hostname}

14

Home EMS sends the following response with two results. The PEV stops executing the first event (/edev/3/frp/1) because it has been superseded by the second event (/edev/3/frp/2). The PEV starts to execute the second event (/edev/3/frp/2) because the time is equal to or greater than the start time.

```
HTTP/1.1 200 OK
Content-Type: application/sep+xml

<FlowReservationResponseList xmlns="urn:ieee:std:2030.5:ns" all="2"
href="/edev/3/frp" results="2" subscribable="1">
  <FlowReservationResponse href="/edev/3/frp/1" subscribable="1">
    <mRID>f8afa6fde40db98d0000ea75</mRID>
    <description>Discharge between 5PM and 12AM</description>
    <!-- 11/8/2013 4:59:50 PM EST -->
    <creationTime>1383947990</creationTime>
    <EventStatus>
      <!-- Superseded-->
      <currentStatus>4</currentStatus>
      <dateTime>1383948300</dateTime>
      <potentiallySuperseded>true</potentiallySuperseded>
    </EventStatus>
    <interval>
      <duration>600</duration>
      <!-- 11/8/2013 5:00 PM EST -->
      <start>1383948000</start>
    </interval>
    <energyAvailable>
      <multiplier>0</multiplier>
      <value>-9623</value>
    </energyAvailable>
    <powerAvailable>
      <multiplier>0</multiplier>
      <value>400</value>
    </powerAvailable>
    <subject>68512866203db3b10000e566</subject>
  </FlowReservationResponse>
  <FlowReservationResponse href="/edev/3/frp/2" subscribable="1">
    <mRID>f8afa6fde40db98d0000ea76</mRID>
    <description>Discharge between 5PM and 12AM</description>
    <!-- 11/8/2013 5:04:50 PM EST -->
    <creationTime>1383948290</creationTime>
    <EventStatus>
      <!-- Active-->
      <currentStatus>1</currentStatus>
      <dateTime>1383948300</dateTime>
      <potentiallySuperseded>false</potentiallySuperseded>
    </EventStatus>
    <interval>
      <duration>600</duration>
      <!-- 11/8/2013 5:05 PM EST -->
      <start>1383948300</start>
    </interval>
    <energyAvailable>
      <multiplier>0</multiplier>
      <value>-9623</value>
    </energyAvailable>
    <powerAvailable>
      <multiplier>0</multiplier>
      <value>500</value>
    </powerAvailable>
    <subject>68512866203db3b10000e566</subject>
  </FlowReservationResponse>
```

Step	Description
	`</FlowReservationResponseList>`

15	PEV provides update of Power Status to end-device server.
	
	`PUT /edev/3/ps HTTP/1.1`
	`Host: {hostname}`
	
	`<PowerStatus xmlns="urn:ieee:std:2030.5:ns">`
	`  <batteryStatus>4</batteryStatus>`
	`  <!-- 11/8/2013 5:05 PM EDT -->`
	`  <changedTime>1383948300</changedTime>`
	`  <!-- mains -->`
	`  <currentPowerSource>1</currentPowerSource>`
	`  <estimatedChargeRemaining>6477</estimatedChargeRemaining>`
	`  <PEVInfo>`
	`    <chargingPowerNow>`
	`      <multiplier>0</multiplier>`
	`      <value>-500</value>`
	`    </chargingPowerNow>`
	`    <energyRequestNow>`
	`      <multiplier>0</multiplier>`
	`      <value>10302</value>`
	`    </energyRequestNow>`
	`    <maxForwardPower>`
	`      <multiplier>0</multiplier>`
	`      <value>6500</value>`
	`    </maxForwardPower>`
	`    <minimumChargingDuration>6906</minimumChargingDuration>`
	`    <targetStateOfCharge>10000</targetStateOfCharge>`
	`    <!-- 9/23/2013 8:00 AM EDT -->`
	`    <timeChargeIsNeeded>1384002000</timeChargeIsNeeded>`
	`    <timeChargingStatusPEV>1383948300</timeChargingStatusPEV>`
	`  </PEVInfo>`
	`</PowerStatus>`

16	**11/8/2013 17:06 EST (1383948360) - PEV Polls Each Minute**
	
	`GET /edev/3/frp?l=2 HTTP/1.1`
	`Host: {hostname}`
	
	EMS Responds with same response as earlier with two events and it has already been acted on by PEV. Step repeated at 17:07, 17:08, and 17:09.

17	**11/8/2013 17:09:50 EST (1383948590) - Third Response Posted by EMS** Home EMS computes and posts next flow reservation response. This is the third response in the sequence so far. The server removes the first response.

18	**11/8/2013 17:10 EST (1383948600) - Third Response Retrieved by PEV**
	PEV sends the following request at 11/8/2013 5:10 PM:
	
	`GET /edev/3/frp?l=2 HTTP/1.1`
	`Host: {hostname}`