The IDM JSON Schema

V.1.32, 2025-09-24

Contents

The IDM JSON Schema

1 Introduction

(Brian's introduction)

Whilst many parameter values are optional, and may simply be omitted, certain parameters are considered essential, in that if missing, the data is essentially useless. These parameters are highlighted with an asterisk.

2 Scope of the IDM Standard

(Brian's material)

The standard captures sufficient product information to:

- Populate a microgrid component product database with manufacturers' product data
- Enable the creation of IT systems and websites to:
    - Select a suitable product, based on known criteria
    - Compare products from different manufacturers
    - Do basic electrical compatibility checking with a view to connecting Port X on component A with Port Y on component B
    - Do basic electrical compatibility checking of several components connected in parallel
    - In cases where the V-I characteristics of two ports are known, calculate the actual voltage and current when they are connected together

The accompanying JSON schema files enable the validation (using any JSON validator) of any microgrid component JSON data against the IDM standard.

The IDM Standard does NOT currently support:

- Modelling the behavior of any microgrid component, beyond what may be observed at a port (although some critical parameters that will be required for this – for example, battery energy capacity – are already included in some component data structures)
- Determination of system power flows
- Determination of system energy balance over time
- Validation of circuit protection measures against a list of fault scenarios

3 Goals of the IDM Standard

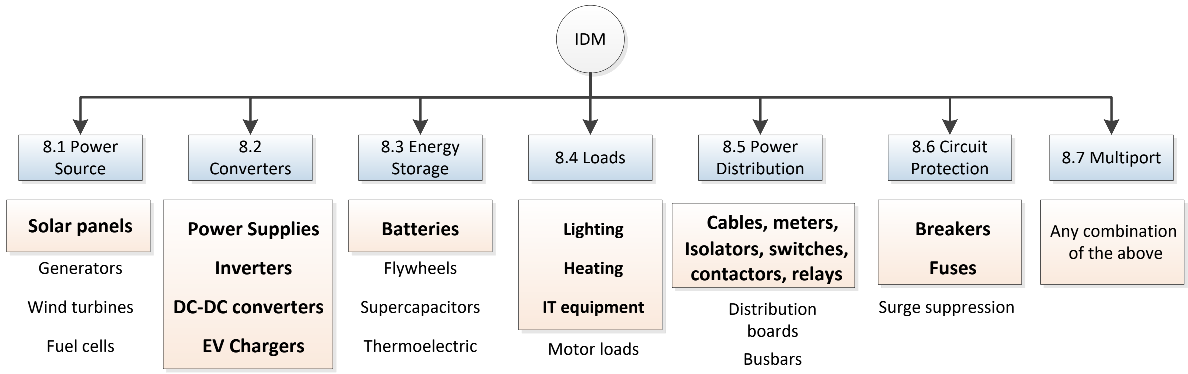
(Brian's material)

3.1 Structure of this document

Sections 1-6...(Brian to add)

Section 7 defines several sub-structures that are common to several microgrid component data structures.

Section 8 provides data structures for most common microgrid components, organized as follows:



(Component types that are not highlighted are to follow in the next version of this document.)

Section 9 discusses how these data structures might be populated

Section 10 discusses how the data might be used.

4 Related Documents

(Brian's material)

5 Terms and Abbreviations

(Brian's material)

Add: **RoHS** - Restriction of Hazardous Substances – a European Union directive that restricts the use of certain hazardous materials in electrical and electronic equipment (EEE) to protect the environment and human health. The directive limits the use of substances like lead, mercury, cadmium, and others in EEE. RoHS compliance is important for manufacturers and importers of electronic products sold within the EU. Most significantly, it requires the use of lead-free solder in manufacturing.

6 Overview

(Brian's material)

6.1 Key Concepts

6.1.1 Active and Passive Components

6.1.1.1 Passive Components

Passive equipment (cables, switches, fuses etc) provides a connection between active components, and may impose limitations, but does not in any way control voltage or current, except possibly to turn it off completely.)

The ports on passive equipment have no defined relationship between voltage and current. They will inherit the characteristics of whatever is connected to their ports. However, they still have a Safe Operating Area. Two-Port Passive devices (TPPD) share a data structure defined in §7.2.21.

6.1.1.2 Active Components

Active electrical/electronic equipment manipulates electrical power flow to achieve an objective.

Active electrical/electronic equipment has one or more ports that allow it to connect to other active equipment (perhaps via some passive devices). An example of this is converters and inverters (§8.2).

6.1.1.3 Smart Components

Smart components are a subset of Active Components, where it is not possible to infer how power will flow, based solely on a knowledge of the electrical conditions of the ports, and the environment in which the component is operating. Smart components have internal states that may vary over time in order to achieve some objective, and which will only be visible to any external system attempting to determine power flows through digital communication.

Put another way, if a smart component behaves on one occasion in a certain way when external electrical conditions are applied to it, it may not behave the same way on some subsequent occasion, due to its (invisible) internal states being different.

6.1.1.4 Limiting situations

IDM only considers the maximum and minimum values of electrical parameters that could occur, to confirm that they are within permissible limits, and therefore when smart components are involved, it cannot determine the actual power and energy flows.

6.1.2 Parameters

For each product, a JSON file contains the applicable parameter values.

The data includes type-specific parameters, using the terminology usually used in manufacturers' datasheets and technical specifications for this type of component (for example, V_{oc}, openCircuitVolts, the open-circuit voltage of a solar panel).

The data also includes parameters in a standardized form, using the same terminology for all component types (for example, maxVoltage). This is included in the Port definitions, which include a Safe Operating Area (SOA), giving the operational limits that the port can tolerate. By standardizing the terminology, it becomes possible to automate the determination of electrical compatibility, as described in §10.2.

6.1.3 Ports

The external connectors on a microgrid component that may be used to connect the component to another component are referred to in this document as “ports”.

Before a port on one component is connected to the port on another component, it is essential to check that they are both electrically and physically compatible.

A port has:

- One or more connectors connected in parallel, possibly individually switched, but sharing a common total current limit
- A number of conductors (“poles”) for the transmission of power
- Possibly some additional poles for data, but not carrying significant power
- It *may* have a Ground (Protective Equipotential, PE) conductor
- A Safe Operating Area (limits to voltage and current in or out)
- It *may* have a defined voltage-current (V-I) relationship for the power conductors

6.2 The format of the schemas

The data structures defined below are available in JSON Schema files stored on GitHub. The files are compliant with the **draft-07** version of the JSON Schema. When used, since they refer to each other, they should all be in the same sub-directory. They may be used with any JSON Validator to confirm that a product JSON data file is compliant with the IDM format.

Where a parameter may have a small number of possible values, and it is unlikely that additional values will ever be required, the possible values are listed in an **enum** array.

6.2.1 Options for data entry

In addition to the **schema** section itself, several of the files include suggestions for the dropdown list of options for entering the parameter, in a **dataEntry** section. This has the format for each possible value:

(short code,
dropdown option as seen by the user,
detailed text explanation of the value available under a ⓘ symbol)

It is of critical importance that a process is defined, whereby additional permissible values for any parameter may be added where genuinely necessary.

6.2.2 Fields for selection tools

Some of the JSON schema files also include a **selectionTool** section, containing suggestions for whether the parameter is appropriate for inclusion in either the filter, display or comparison section of a selection tool (see §10.1for how these might be used).

6.2.1 Summing for connecting devices in parallel

When two ports are connected together, for the purposes of connecting a third device in parallel, they may be considered as a single port, with a new set of parameters. For example, a constant-voltage power supply connected to a load may be considered for the purposes of connecting another load to still be a constant-voltage power supply, but with a reduced current capacity. Likewise, a grounded supply remains a grounded supply when an ungrounded load is attached. The new effective parameter value when two ports are connected is defined in a **summing** section (*value from port A, value from port B, resulting value*) – ports A and B can be either way round.

Currently, only the "textList" method is supported. Two text strings result in a third text string. (Exceptionally, if the table shows an empty array element, then (and only then), if the two text strings are identical, the produced text string will be the same. This is just to save lots of [[A,A,A],[B,B,B],...] entries.)

In some summing cases, arithmetic will be required to calculate a new numerical value. Application of diversity may also be necessary (for example, to sum peak power ratings). This will require additional notation, and is for further study.

By successively combining two ports, any number of devices may be connected in parallel.

If the summing table does not have an entry giving the resultant value when checked either way round (A,B and B,A), then the two ports are incompatible.

7 Shared Data Structures

7.1 Universal JSON Parameters

File: `common-schema.json`

The parameters which are common to all products in the catalog are defined in this file. Listed below are all the parameters that are common to all types of product. Particular types of product have their own schemas defined in separate files, which must be in the same folder, or sub-folders.

7.1.1 Manufacturer *

File: `manufacturer-schema.json`

Information about the manufacturer of the product, ie where to go for the most authoritative source of information about the product.

7.1.1.1 JSON Schema

This parameter uses the standard JSON format for describing a company or other organization (see §7.2.1).

7.1.1.2 Selection tool presentation

The company name appears in both the selection filter and the selection display, with a hyperlink to the company's main website home page.

7.1.2 Product Name

File: `productName-schema.json`

A name given to the product by the manufacturer.

7.1.2.1 JSON Schema

```
{
	"schema": {
		"type": "string",
		"minLength": 2,
		"maxLength": 254
	}
}
```

7.1.2.2 Selection tool presentation

The product name appears in the selection filter.

7.1.3 Product Identifier *

File: `productIdentifier-schema.json`

The part number assigned to the product by the manufacturer.

7.1.3.1 JSON Schema

```
{
	"schema": {
		"type": "string",
		"minLength": 2,
		"maxLength": 254
	}
}
```

7.1.3.2 Selection tool presentation

The product name appears in the selection display.

7.1.4 Product Series

File: `productSeries-schema.json`

The product range, family or series that this product forms part of, if any.

7.1.4.1 JSON Schema

```
{
	"schema": {
		"type": "string",
		"minLength": 0,
		"maxLength": 254
	},
	"selectionTool": {
		"filter": true,
		"display": false
	}
```

7.1.4.2 Selection tool presentation

The product series only appears in the selection filter.

7.1.5 Datasheet Hyperlink

File: `common-schema.json`

This should be a specific hyperlink either to the manufacturer's product web page (preferred), or to a downloadable product manual.

(Note that the schema also supports the uploading of any number of files by the manufacturer.)

7.1.5.1 JSON Schema

```
{
        "schema": {
                "type": "string",
                "format": “uri”
        }
}
```

7.1.5.2 Selection tool presentation

This is used as a hyperlink behind the thumbnail image of the product in the selection tool display section.

7.1.6 Description

File: `common-schema.json`

This is a free text field in which a prose description of the product may be provided.

7.1.6.1 JSON Schema

```
{
   "description": {
      "type": "string",
      "minLength": 0,
      "maxLength": 65535
    }
}
```

7.1.6.2 Selection tool presentation

This is not displayed unless on a product-specific web page hyperlinked from the display row of the selection tool.

7.1.7 Distributors

File: `distributors-schema.json`

A list of distributors, wholesalers or importers of the (commodity) product. This may be a single company, or an array of companies, each using the company schema format defined in §7.2.1.

It is assumed that this will be populated by the manufacturer, to provide pointers to companies likely to hold stock.

7.1.7.1 JSON Schema

```
{
   "schema": {
    "anyOf": [
      {"$ref": "company-schema.json#/schema"},
```

```
      {
        "type": "array",
        "items": {"$ref": "company-schema.json#/schema"}
      }
    ]
  },
  "selectionTool": {
    "filter": false,
    "display": true
  }
}
```

7.1.7.2 Selection tool presentation

These may be listed (and potentially hyperlinked) from the display row of the selection tool.

7.1.8 Systems Integrators

File: `distributors-schema.json` (We can use the same file for this)

A list of companies that are approved systems integrators for complex products. This may be a single company, or an array of companies, each using the company schema format defined in §7.2.1.

7.1.8.1 JSON Schema

```
{
   "schema": {
    "anyOf": [
      {"$ref": "company-schema.json#/schema"},
      {
        "type": "array",
        "items": {"$ref": "company-schema.json#/schema"}
      }
    ]
  },
  "selectionTool": {
    "filter": false,
    "display": true
  }
}
```

7.1.8.2 Selection tool presentation

These may be listed (and potentially hyperlinked) from the display row of the selection tool.

7.1.9 Embargoed countries

File: `embargoedCountries-schema.json`

There may be certain end-use countries to which under US law it is not permitted to ship products. These may be listed here, using the ISO-3166-1 A-2 two-letter coding.

7.1.9.1 JSON Schema

```
{
  "description": "List countries to which it is forbidden by law to ship this
product.",
  "schema": {
    "oneOf": [
```

```
              {
                "type": "string",
                "minLength": 2,
                "maxLength": 2
              },
              {
                "type": "array",
                "items": {
                  "type": "string",
                  "minLength": 2,
                  "maxLength": 2
                }
              }
            ]
          }
```

7.1.10 Not recommended for new designs

File: `common-schema.json`

If a product is End-of-Life, it should not be included in new designs, although it may still be available for replacement purposes. This is a Boolean yes/no flag.

7.1.10.1 JSON Schema

```
{
  "schema": {
    "type": "boolean”
  }
}
```

7.1.10.2 Selection tool presentation

It is suggested that if a product is EoL, it is either not displayed at all in the display section of the selection tool, or that it is greyed out.

7.1.11 Type-specific Parameters

File: `typeSpecifics-schema.json`

Each type of microgrid component has critical parameters that are specific to its function. These are listed separately by product type in §8.

7.1.12 Listing Authorities

File: `listingAuthorities-schema.json`

A list of organizations that have certified the product to meet certain standards. Each organization can create a profile in the format defined in §7.2.1.

7.1.12.1 JSON Schema

```
{
   "schema": {
    "anyOf": [
      {"$ref": "company-schema.json#/schema"},
      {
```

```
          "type": "array",
          "items": {"$ref": "company-schema.json#/schema"}
        }
      ]
    },
    "selectionTool": {
      "filter": false,
      "display": "<a href \"https:$webHomePageURL$\">$coLogo$</a>"
    }
  }
```

7.1.12.2 Suggested initial dropdown list

This might include: UL, CSA, TUV, Nemko, CE, UKCA – though each will need an organization profile JSON file.

7.1.12.3 Selection tool presentation

These may be listed (and potentially hyperlinked) from the display row of the selection tool. Icons may be used in preference to organization names. Where a specific test reference is available (as for example provided by SEC in Chile), this can be hyperlinked under the icon.

7.1.13 Environmental Parameters

File: `environmental-schema.json`

The manufacturer may declare certain environmental constraints on the safe and reliable operation of the product. These are listed in §7.2.4.

7.1.14 Files

File: `files-schema.json`

The manufacturer may upload various product-specific files, which the microgrid designer can download.

The filename extension will normally indicate how the file should be handled.

A filetype is also included. It is suggested that this should give the IANA-defined MIME type, eg “image/jpeg” or “application/pdf”, etc.

7.1.14.1 Schema

```
{
  "schema": {
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "filename": { "type": "string" },
        "fileType": { "type": "string" }
      }
    }
  }
}
```

7.1.15 Images

File: `images-schema.json`

The manufacturer may upload various pictures of the product, which the microgrid designer can view.

7.1.15.1 Schema

```
{
  "schema": {
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "filename": {"type": "string"},
        "mimeType": { "type": "string" }
      }
    }
  }
}
```

7.1.16 List Price

The list price in US Dollars is not intended to represent the price that will actually apply, but is merely included to allow product comparison based on price. It may be omitted if the manufacturer feels it might be contentious.

7.2 Shared Parameter Definitions

There are several parameters that appear in more than one context (for example, current limits may be rated currents, peak currents or breaking currents, but the format for defining a maximum current may be the same).

7.2.1 Company/Organization Profile Data Format

File: `company-schema.json`

Any organization involved in IDM can create an extensive profile for themselves, using the profile model developed for DC-IDE, including a logo. This will significantly extend this part of the schema. It should be a separate JSON file, referenced by the product-specific JSON file.

7.2.1.1 JSON Schema

The standard JSON format for describing a company or other organization is:

```
{
  "schema": {
    "type": "object",
    "properties": {
      "coName": {
        "type": "string",
        "minLength": 3,
        "maxLength": 254
      },
      "webHomePageURL": {
        "type": "string",
        "format": "uri"
      }
    },
    "$comment": "The logo can easily be represented here by a Base64 string",
    "coLogo": {"type": "string"},
    "required": [ "coName", "webHomePageURL"]
  },
  "selectionTool": {
    "display": "<a href \"https:$webHomePageURL$\">$coName$</a>",
    "$comment": "This will need some coding to build the hyperlink"
  }
}
```

The amount of company material to include in the IDM data structure is for discussion, particularly as the company is free to upload as many files as it wishes.

7.2.2 Mechanical Attributes

File: `mechanical-schema.json`

These comprise size, weight and mounting styles.

7.2.2.1 Schema

```
"schema": {
  "type": "object",
  "properties": {
    "length": {"$ref": "dimensions-schema.json#/schema"},
    "width": {"$ref": "dimensions-schema.json#/schema"},
    "depth": {"$ref": "dimensions-schema.json#/schema"},
    "height": {"$ref": "dimensions-schema.json#/schema"},
    "diameter": {"$ref": "dimensions-schema.json#/schema"},
```

```
        "size": {
          "type": "object",
          "properties": {
            "length": {
              "type": "number",
              "minimum": 0
            },
            "width": {
              "type": "number",
              "minimum": 0
            },
            "depth": {
              "type": "number",
              "minimum": 0
            },
            "height": {
              "type": "number",
              "minimum": 0
            },
            "diameter": {
              "type": "number",
              "minimum": 0
            },
            "units": {
              "type": "string",
              "enum": [
                "in",
                "ft",
                "yds",
                "mi",
                "mm",
                "cm",
                "m",
                "km"
              ]
            }
          },
          "required": ["units"]
        },
        "weight": {
          "oneOf": [
            {
              "type": "object",
              "properties": {
                "value": {"type": "number"},
                "units": {
                  "type": "string",
                  "enum": [
                    "g",
                    "kg",
                    "oz",
                    "lbs"
                  ]
                }
              },
              "required": ["value"]
            },
            {
              "type": "string",
              "pattern": "^([0-9]+(.[0-9]+)?)((m|k)?g|oz|lbs)$"
            }
          ]
```

```
        },
        "mountingType": {
          "oneOf": [
            {"$ref": "mountingStyle-schema.json#/schema"},
            {
              "type": "array",
              "items": {"$ref": "mountingStyle-schema.json#/schema"}
            },
            {
              "type": "array",
              "items": {
                "type": "object",
                "properties": {
                  "style": {"$ref": "mountingStyle-schema.json#/schema"},
                  "orderCode": {"type": "string"}
                }
              }
            }
          ]
        }
      }
    }
```

7.2.3 Physical Dimensions

File: `dimensions-schema.json`

All measurements of length can adhere to the same schema.

Dimensions may be entered as a JSON object {“Value”,”Units”) or as a string comprising numbers followed by the units, eg:

```
"diameter": {
    "value": 12.5,
    "units": "mm"
}
```

...or:

```
"diameter": "12.5mm",
```

Either format is equally valid. (As shown in §7.2.2.1 above, it is also possible to list physical dimensions as numbers without units, and provide a single common units value for all the numbers.)

7.2.3.1 JSON Schema

```
{
  "$comment": "This schema may be used for any physical length value",
  "schema": {
    "oneOf": [
      {
        "type": "object",
        "properties": {
          "value": { "type": "number" },
          "units": {
            "type": "string",
            "enum": [ "in", "ft", "yds", "mi", "mm", "cm", "m", "km" ]
          }
        },
```

```
        "required": [ "value", "units" ]
      },
      {
        "type": "string",
        "pattern": "^([0-9]+(.[0-9]+)?)(in|ft|yds|mi|mm|cm|m|km)$"
      }

    ]
  }

}
```

7.2.4 Environmental Data

File: `environmental-schema.json`

The manufacturer may declare certain environmental constraints on the safe and reliable operation of the product. Specifically, this may relate to:

- Ambient operating temperature range,
- Operating Relative Humidity (as defined in IEC 60068-2-11)
- Ingress protection standards (IEC60529 and/or NEMA)
- Operating altitude
- Cooling method
- RoHS Compliance

7.2.4.1 JSON Schema

```
"schema": {
  "type": "object",
  "properties": {
    "operatingTemperature": {
      "type": "object",
      "properties": {
        "min": {"type": "number"},
        "max": {"type": "number"},
        "unit": {
          "type": "string",
          "enum": [
            "C",
            "F"
          ]
        }
      },
      "additionalProperties": false
    },
    "operatingHumidity": {
      "type": "object",
      "$comment": "IEC 60068-2-11",
      "properties": {
        "min": {
          "type": "number",
          "minimum": 0
        },
        "max": {
          "type": "number",
          "maximum": 100
        }
      },
      "additionalProperties": false
```

```
        },
        "ingressProtection_IP": {
          "type": "string",
          "pattern": "^IP([0-6])|x([0-9])|x[ABCD]*[HMSW]*$"
        },
        "ingressProtection_NEMA": {
          "type": "string",
          "enum": [
            "1",
            "2",
            "3",
            "3X",
            "3S",
            "3SX",
            "3R",
            "3RX",
            "4",
            "4X",
            "5",
            "6",
            "6P",
            "12",
            "12K",
            "13"
          ]
        },
        "maximumOperatingAltitude": {"$ref": "dimensions-schema.json#/schema"},
        "coolingMethod": {
          "type": "string",
          "enum": [
            "passive",
            "forced-air",
            "liquid",
            "none"
          ]
        },
        "RoHScompliant": {"type": "boolean"},
        "DomesticComponentRequirement": {"type": "boolean"}
      }
           }
```

7.2.5 Current Rating

File: `currentRating-schema.json`

Current limits are specified at several points in the product schemas. Currents may be entered as a JSON object {“Value”,”Units”) or as a string comprising numbers followed by the units, eg:

```
              "currentRating": {
                  "value": 10,
                  "units": "A"
              }
```

...or:

```
              "currentRating": "1.5kA",
```

Either format is equally valid.

7.2.5.1 JSON Schema

```
{
  "$comment": "This schema may be used for any component current rating",
  "schema": {
    "oneOf": [
      {
        "type": "object",
        "properties": {
          "value": {"type": "number"},
          "units": {
            "type": "string",
            "enum": [ "mA", "A", "kA"]
          }
        },
        "required": [ "value", "units"]
      },
      {
        "type": "string",
        "pattern": "^([0-9]+(.[0-9]+)?)(k|m)?A$"
      }
    ]
  }
}
```

7.2.6 Power Rating

File: `powerRating-schema.json`

Power ratings are specified at several points in the product schemas. Powers may be entered as a JSON object {“Value”,”Units”) or as a string comprising numbers followed by the units, eg:

```
"powerRating": {
    "value": 10,
    "units": "W"
}
```

...or:

```
"powerRating": "1.5kW",
```

Either format is equally valid.

7.2.6.1 JSON Schema

```
"schema": {
    "oneOf": [
      {
        "type": "object",
        "properties": {
          "value": { "type": "number" },
          "units": {
            "type": "string",
            "enum": [ "mW", "W", "kW" ]
          }
        },
        "required": [ "value", "units" ]
      },
      {
        "type": "string",
        "pattern": "^(-?[0-9]+(.[0-9]+)?)(k|m)?W$"
```

```
        }
      ]
```

7.2.7 Voltage Rating

File: `voltageRating-schema.json`

Voltage limits are specified at several points in the product schemas. Voltages may be entered as a JSON object {“Value”,”Units”) or as a string comprising numbers followed by the units, eg:

```
"voltageRating": {
    "value": 600,
    "units": "mV"
}

```

...or:

```
"voltageRating": "1.25kV",

```

Either format is equally valid. (As shown in §7.2.18.1 below, it is also possible to voltages as numbers without units, and provide a single common units value for all the numbers.)

7.2.7.1 JSON Schema

```
{
  "$comment": "This schema may be used for any component voltage rating",
  "schema": {
    "oneOf": [
      {
        "type": "object",
        "properties": {
          "value": {"type": "number"},
          "units": {
            "type": "string",
            "enum": [ "mV", "V", "kV"]
          }
        },
        "required": [ "value", "units"]
      },
      {
        "type": "string",
        "pattern": "^([0-9]+(.[0-9]+)?)(k|m)?V$"
      }
    ]
  }
}

```

7.2.8 Resistance Values

File: `resistanceValue-schema.json`

Resistances are specified at several points in the product schemas. They may be entered as a JSON object {“Value”,”Units”) or as a string comprising numbers followed by the units, eg:

```
"resistanceValue": {
    "value": 600,
    "units": "milliohms"
}
```

...or:

```
"resistanceValue": "1.25kΩ",
```

Either format is equally valid. The permissible values for the units of resistance are: "milliohms", "ohms", "kiloohms. megohms", "m\u03A9", "\u03A9", "k\u03A9", "M\u03A9", "mΩ", "Ω", "kΩ", "MΩ" . (\u03A9 is the Unicode value of the omega symbol.)

7.2.8.1 JSON Schema

```
  "$comment": "This schema may be used for any resistance.  Note unicode points may
need different syntax, depending on usage",
  "schema": {
    "oneOf": [
      {
        "type": "object",
        "properties": {
          "value": { "type": "number" },
          "units": {
            "type": "string",
            "enum": [ "milliohms", "ohms", "kiloohms. megohms", "m\u03A9", "\u03A9",
"k\u03A9", "M\u03A9", "mΩ", "Ω", "kΩ", "MΩ" ]
          }
        },
        "required": [ "value", "units" ]
      },
      {
        "type": "string",
        "pattern": "^([0-9]+(.[0-9]+)?)(m|k|M)?(ohms|Ω|\u03A9)$"
      }

    ]
  }
```

7.2.9 Connection type

File: `phase-schema.json`

Connections may be single-, split- or three-phase AC, or DC. Three-phase AC may be Delta (3 wires) or “Y” (4 wires, including a Neutral). The latter may also be used to power single-phase AC loads.

(Currently, bipolar DC with an extended midpoint conductor is not a supported configuration under IDM.)

The permissible values are:

7.2.9.1.1 Schema values

```
[
      ["D3S","Delta 3ph power","The port provides a 3-wire 3-phase power source"],
      ["D3L","Delta 3ph load","The port provides a 3-wire 3-phase load (eg 208VAC or
415VAC)"],
      ["Y4S","Y 3+N power","The port provides a 4-wire 3-phase+Neutral power source”],
      ["Y4L","3+N Loads","The load(s) need 3phase and Neutral"],
      ["P1S","AC 1ph power","Single phase - The power source just has Live and
Neutral"],
```

```
	["P1L","AC 1ph load","Single-phase - The load just has Live and Neutral (eg 120VAC or 220-240VAC)"],
	["P2S","AC split-phase supply","The supply has two opposite phases, eg 120/240V. Neutral is also available"],
	["P2L","AC split-phase load","The load needs two live phases at 180 degrees, eg 120/240VAC (no Neutral connection)"],
	["DC","2-wire DC","The port has Positive and Negative"]
]
```

7.2.10 Frequency

Every port has an expectation that it will have DC of a certain polarity, or AC of a certain frequency or range of frequencies on it. A table may be made of the various possibilities, with an abbreviated code for each option. The options are:

7.2.10.1.1 Possible values

```
	[ "DC,", "DC", "The port expects or delivers Direct Current" ],
	[ "P50", "Provides 50Hz", "The port defines the frequency as 50Hz.  No connected port may also define the frequency." ],
	[ "P60", "Provides 60Hz", "The port defines the frequency as 60Hz.  No connected port may also define the frequency." ],
	[ "S50", "Syncs to 50Hz", "If 50Hz is present, this will synchronise with it, and if not, it will create 50Hz." ],
	[ "S60", "Syncs to 60Hz", "If 60Hz is present, this will synchronise with it, and if not, it will create 60Hz." ],
	[ "S5060", "Syncs to 50Hz or 60Hz", "If 50Hz or 60Hz is present, this will synchronise with it, but if absent, it will not create it." ],
	[ "N50", "Needs 50Hz", "The port expects a nominally 50Hz supply." ],
	[ "N60", "Needs 60Hz", "The port expects a nominally 60Hz supply." ],
	[ "N5060", "Needs 50-60Hz", "The port expects a supply with a frequency of nominally either 50Hz or 60Hz." ],
	[ "N50DC", "50Hz or DC", "The load or passive device will work with either 50Hz AC or DC" ],
	[ "N60DC", "60Hz or DC", "The load or passive device will work with either 60Hz AC or DC" ],
	[ "N5060DC", "50Hz, 60Hz or DC", "The load or passive device will work with either 50/60Hz AC or DC" ]
```

When two ports are connected together, these values must be taken into account, producing a new frequency value, which must be used if a third component is connected in parallel. The permissible combinations are:

7.2.10.1.2 Permissible combinations, and the net result

(portA value), (portB value), (resulting value when connected)

```
	[ "DC", "DC", "DC" ],
	[ "DC", "N5060DC", "DC" ],
	[ "P50", "N50", "P50" ],
	[ "P60", "N60", "P60" ],
	[ "P50", "S50", "P50" ],
	[ "P60", "S60", "P60" ],
	[ "P50", "S5060", "P50" ],
	[ "P60", "S5060", "P60" ],
	[ "P50", "N5060", "P50" ],
	[ "P60", "N5060", "P60" ],
	[ "P50", "S5060DC", "P50" ],
	[ "P60", "S5060DC", "P60" ],
	[ "P50", "N5060DC", "P50" ],
	[ "P60", "N5060", "P60" ],
	[ "N50", "N50", "N50" ],
```

```
        [ "N50", "N50", "N50" ],
        [ "N60", "N60", "N60" ],
        [ "N50", "N5060", "N50" ],
        [ "N60", "N5060", "N60" ],
        [ "N50", "N5060DC", "N50" ],
        [ "N60", "N5060DC", "N60" ],
        [ "N5060DC", "N5060DC", "N5060DC" ]
```

These are all defined in `frequency-schema.json`.

7.2.11 Grounding and Polarity

File: `grounding-schema.json`

In DC and mixed AC-DC systems, it is important that there is only one ground connection in the system, or if more than one, all but one must be connected via diodes to give a forward voltage drop that is larger than any possible ohmic voltage drop. This is to avoid spurious ground currents that would cause electrolytic corrosion. Pure AC systems are allowed to have multiple grounds (known as TN-CS or PME).

If a microgrid component connects either power rail to ground or to a Protective Equipotential (PE) line, it makes it important that whatever it is connected to does not. (If it did, current would flow in the PE line, which is unacceptable except in the event of a fault.)

7.2.11.1.1 Schema

```
      "description": "If the component includes a connection between either power
    pole and the earth or a PE conductor",
      "mandatory": true,
      "schema": {
        "type": "string",
        "enum": [ "Pos", "Neg", "Float", "Live", "Neut", "PVD", "NVD" ]
      },
      "dataEntry": {
        "prompt": "Is a power pole connected to a local ground?",
        "entryType": "dropdown",
        "dropdownValues": [
          [ "Pos", "Positive grounded", "The positive pole is grounded, the live
    has a negative voltage" ],
          [ "Neg", "Negative grounded", "The negative pole is grounded, the live
    has a positive voltage" ],
          [ "Float", "No pole grounded", "Neither power pole is grounded in this
    component" ],
          [ "Live", "Both power poles live", "The component has a local ground, and
    imposes some voltage on both power poles" ],
          [ "Neut", "AC Neutral grounded", "The AC Neutral is grounded" ],
          [ "PVD", "Positive via diodes", "The positive pole is grounded locally
    via diodes" ],
          [ "NVD", "Negative via diodes", "The negative pole is grounded locally
    via diodes" ]

        ]
      },
      "summing": {
        "method": "textList",
        "list": [
          [ "Pos", "Float", "Pos" ],
          [ "Neg", "Float", "Neg" ],
          [ "Neut", "Float", "Neut" ],
          [ "Pos", "PVD", "Pos" ],
          [ "Neg", "NVD", "Neg" ],
```

```
            [ "Neut", "Neut", "Neut" ],
            [ "Live", "Float", "Live" ],
            [ "PVD", "Float", "PVD" ],
            [ "NVD", "Float", "NVD" ],
            [ "Float", "Float", "Float" ]

          ]
        }
      }
```

7.2.12 Ground Wire

File: `PE-schema.json`

A Ground Wire (aka Protective Equipotential Wire, PE) may be offered by a port, or required by a port, or neither.

7.2.12.1.1 Schema

```
      {
        "description": "If the port offers or requires a Protective Equipotential
      (Ground) Wire",
        "mandatory": true,
        "schema": {
          "type": "string",
          "enum": [ "offered","required","neither" ]
        },
        "dataEntry": {
          "prompt": "Is a PE (Ground) wire offered or required?",
          "entryType":  "dropdown",
          "dropdownValues": [
            [ "Offered", "PE Offered", "The port provides a PE (Ground) connection"
      ],
            [ "Required", "Required", "The port requires a PE (Ground) connection" ],
            [ "Neither", "No ground connection", "This port neither offers nor
      requires a PE/Ground wire" ]
          ]
        },
        "summing": {
          "method": "textList",
          "list": [
            [ "", "", "" ],
            [ "Offered", "Required", "Offered" ],
            [ "Offered", "Neither", "Offered" ],
            [ "Required", "Neither", "Required" ]
          ]
        }
      }
```

7.2.13 Mounting Style

File: `mountingStyle-schema.json`

Most microgrid components expect to be fixed to something.

7.2.13.1 Initial Suggested dropdown list of mounting options

- "Surface mount",
- "DIN rail",
- "Panel mount",
- "Wall-mount",

- "Free-standing"

7.2.14 Cable Sizes

File: `wireSizes-schema.json`

Cable sizes have to be specified in several contexts in microgrid design.

7.2.14.1 Initial dropdown values

The standard sizes are:

- "30AWG, 0.05mm2",
- "28AWG, 0.08mm2",
- "26AWG, 0.14mm2",
- "24AWG, 0.25mm2",
- "22AWG, 0.34mm2",
- "21AWG, 0.38mm2",
- "20AWG, 0.50mm2",
- "18AWG, 0.75mm2",
- "17AWG, 1.0mm2",
- "16AWG, 1.5mm2",
- "14AWG, 2.5mm2",
- "12AWG, 4.0mm2",
- "10AWG, 6.0mm2",
- "8AWG, 10mm2",
- "6AWG, 16mm2",
- "4AWG, 25mm2",
- "2AWG, 35mm2",
- "1AWG, 50mm2",
- "1/0AWG, 55mm2",
- "2/0AWG, 70mm2",
- "3/0AWG, 95mm2"

7.2.15 Bolt Sizes

File: `boltSize-schema.json`

There are many different standards for bolt threads. For microgrid purposes, the only important parameter is the required hole size for the eyelet terminating the cable.

7.2.15.1 Initial dropdown values

The standard sizes are:

```
"10-32 (3/16\")",
"12-24 (7/32\")",
"M6 (1/4\")",
"M8 (5/16\")",
"M10 (13/32\")",
"M12 (1/2\")"
```

7.2.16 Mechanical Contacts

File: `mechanicalContact-schema.json`

Switches, isolators, breakers, relay contacts and even plugs and sockets can interrupt the flow of current. Regardless of where the contact is located, the limiting issues are the same. They are:

- Ohmic heating during steady-state current flow (limiting the current)
- Arcing as the current flow is interrupted (limiting the open-circuit voltage)
- The maximum current that can be interrupted (in a fault situation, this may be much higher than the steady-state current)
- Whetting current (this is rarely an issue with power components, but can be critical for control ports)
- Making current – there may be a requirement that there is pre-charge protection, to prevent very high capacitor charging currents when a contact is first closed.

7.2.16.1 Schema

```
"schema": {
  "type": "object",
  "properties": {
    "maxVoltageAC": { "$ref": "voltageRating-schema.json#/schema" },
    "maxVoltageDC": { "$ref": "voltageRating-schema.json#/schema" },
    "maxSteadyCurrent": { "$ref": "currentRating-schema.json#/schema" },
    "ampsBreakingCapacity": { "$ref": "currentRating-schema.json#/schema" },
    "minimumWhettingCurrent": { "$ref":"currentRating-schema.json#/schema" },
    "prechargeProtectionRequired": { "type": "boolean" },
    "configuration": {
      "type": "string",
      "enum": [ "Normally Open", "Normally Closed", "Change-over", "Make-
before-break" ]
    }
  }
}
```

Note that MaxVoltageDC is always positive, and MaxSteadyCurrent and AmpsBreakingCapacity are always positive, regardless of the direction of the current.

7.2.17 Connection Style

File: `connection-schema.json`

Most microgrid components are permanently wired in place via clamping screw terminals (onto wire ends or ferrules) or by eyelets onto bolts. Screw terminals are specified by the wire capacity they can accommodate, bolts by the outside diameter of the bolt, which will require the eyelet to be slightly larger.

7.2.17.1 Connection schema

```
"schema": {
  "oneOf": [
    {
      "type": "object",
      "properties": {
        "connectionType": { "const": "bolt" },
        "bolt": { "$ref": "boltSize-schema.json#/schema" }
      },
```

```
                "required": [ "bolt" ]
              },
              {
                "type": "object",
                "properties": {
                  "connectionType": { "const": "terminal" },
                  "terminal": { "$ref": "wireSizes-schema.json#/schema" }
                },
                "required": [ "terminal" ]
              },
              { "$ref": "#definitions/plugAndSocket" },
              {
                "type": "array",
                "items": { "$ref": "#definitions/plugAndSocket" }
              },
              {
                "type": "object",
                "properties": {
                  "connectionType": { "const": "other" },
                  "other": { "type": "string" }
                },
                "required": [ "other" ]
              }
            ]
          },
          "definintions": {
            "plugAndSocket": {
              "type": "object",
              "properties": {
                "connectionType": { "const": "plugAndSocket" },
                "type": { "type": "string" },
                "gender": { "type": "string" }
              },
              "required": [ "type", "gender" ]
            }
          }
```

Wire sizes are defined in §7.2.14. Bolt sizes are defined in §7.2.15.

7.2.18 Safe Operating Area (SOA)

File: `SOA-schema.json`

Ports on microgrid components are connected to ports on other components, to create a complete microgrid system. The SOA provides a means for determining whether it is possible to connect two ports together without damage.

Note that this does not consider at all what happens *inside* the microgrid component!

SOA parameter values may duplicate values elsewhere in the data structure. For example, a solar panel will have an open-circuit voltage value declared elsewhere, and also have and SOA/maxVoltage parameter, which may have the same value.

The application of SOA values to the determination of compatibility is described in §10.2.

7.2.18.1 Min, Max and Nominal Voltage

The Maximum Voltage is the highest voltage the port should be expected to operate correctly with. It should always be a positive value, regardless of the polarity of the supply.

The Minimum Voltage is the lowest voltage consistent with proper operation. (A voltage of zero will always be acceptable, but nothing will operate.)

Nominal Voltage is the single typical voltage for the product. It is not used to determine the Safe Operating Area.

7.2.18.2 Max Current In/Out

These two parameters refer to the steady-state current. Higher currents may be tolerated for short durations.

The MaxCurrentOut parameter will always be positive (or zero for a power sink). The current flows from the positive supply to the negative, and it implies power flowing OUT of the port. In the case of AC, it is current that is in phase with the voltage.

The MaxCurrentIn parameter will always be negative (or zero for a device that is exclusively a power source). The current flows into the positive supply and out of the negative, and it implies power flowing INTO the port. In the case of AC, it is current that is at 180° to the voltage.

(The IDM model does not currently deal with AC power factors other than unity.)

7.2.18.3 SOA Schema

```
  "$comment": "This JSON schema defines the Safe Operating Area of any electrical
power port",
  "schema": {
    "type": "object",
    "properties": {
      "minVoltage": {"$ref": "voltageRating-schema.json#/schema"},
      "maxVoltage": {"$ref": "voltageRating-schema.json#/schema"},
      "nominalVoltage": {"$ref": "voltageRating-schema.json#/schema"},
      "voltage": {
        "type": "object",
        "properties": {
          "min": {"type": "number"},
          "max": {"type": "number"},
          "nominal": {"type": "number"},
          "units": {
            "type": "string",
            "enum": [
              "mV",
              "V",
              "kV",
              "Vrms",
              "kVrms"
            ]
          }
        },
        "required": ["units"],
        "additionalProperties": false
      },
      "maxCurrentOut": {"$ref": "currentRating-schema.json#/schema"},
      "maxCurrentIn": {"$ref": "currentRating-schema.json#/schema"},
      "maxPowerIn": {"$ref": "powerRating-schema.json#/schema"},
      "maxPowerOut": {"$ref": "powerRating-schema.json#/schema"}
    }
```

```
        }
```

7.2.19 Port V-I Relationship Types

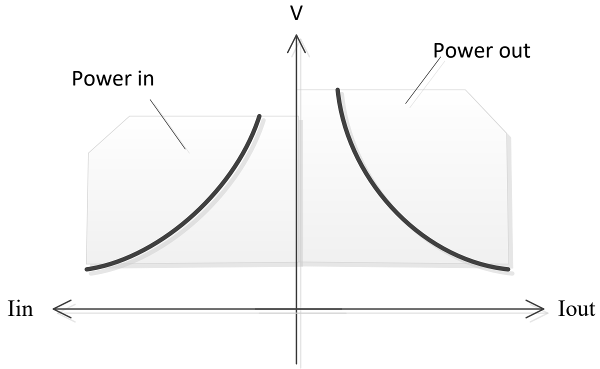
File: `port-schema.json`

In active equipment, one of the three key parameters (voltage, current or power) will be controlled by hardware or software, and the other two will be determined by whatever the port is connected to.

7.2.19.1 Constant power port

Typically, a constant-power port will either be a power source or a power sink, but bidirectional ports are possible. The power ratings in the two directions may not be the same. Most electronic loads exhibit a constant-power load characteristic (though of course the power level may vary with time, as the load does its job).

The V-I characteristic may be represented graphically:



7.2.19.1.1 Voltage Droop Control

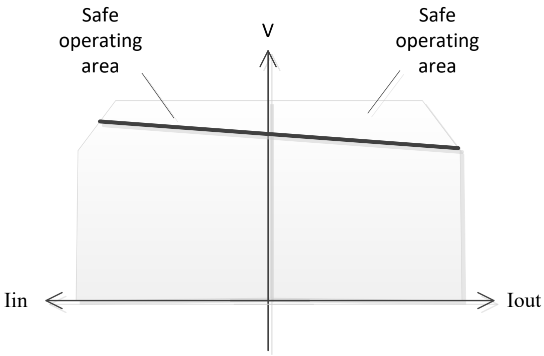
A particular form of constant power control is "voltage droop control", where the voltage observed indicates the level of power required. A voltage higher than nominal indicates that the microgrid has excess power, and that the power being supplied should be reduced. Conversely, lower voltages indicate that more power should be supplied. This may be implemented in hardware or software. Particular attention must be paid to this when more than one power source is feeding a DC bus, to avoid oscillation between supplies.

7.2.19.1.2 Frequency Power Control

The equivalent to voltage droop control in AC microgrids is frequency control – a frequency higher than nominal indicated power over-supply, and below nominal power under-supply.

7.2.19.2 Constant Voltage Port

Most electronic power supplies will exhibit a constant-voltage output characteristic (the output voltage will vary only very slightly between zero and full rated current output).



A constant-voltage port will deliver (or draw in) whatever current will maintain the voltage at the level set by the hardware of the component. (The controlled value may be DC, or single- or polyphase AC, with a constant RMS voltage.) There will be a maximum current determined by the power limitations of the converter, beyond which constant voltage control breaks down.

In practice, there will always be a small variation in terminal voltage as the current varies. The general strategy is to try to minimize this.

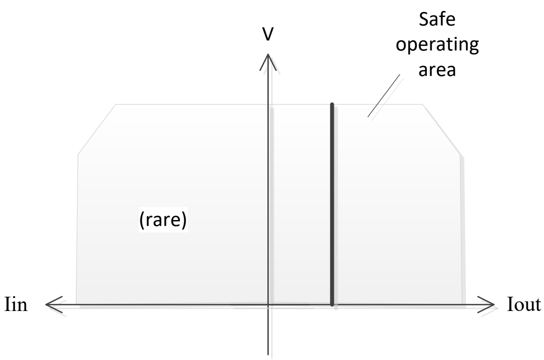
$$V_{out} = (V_{oc} - IR_{thevenin})$$

Typically, a constant-voltage port will either be a power source or a power sink, but bidirectional ports are possible. The Thevenin resistance will be small in comparison with the resistance of any load.

Exceptionally, the target constant voltage V_{oc} may be determined by manual adjustment (in for example, a lab bench power supply) or as in the case of USB, by digital communication and software.

7.2.19.3 Constant Current

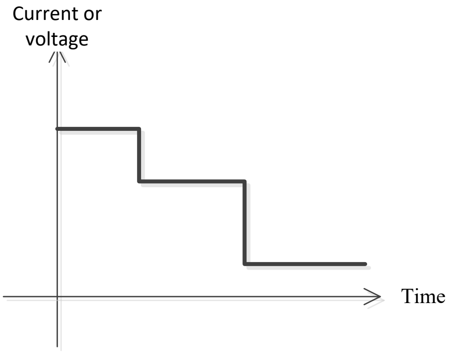
A constant-current port will either source or sink the current at a level specified by the hardware. If sourcing current, the voltage will be set to deliver the required current. There will be a maximum voltage the supply will support in trying to maintain the specified current (for example into an open circuit).



Constant-current supplies are important for delivering uniform brightness in LED lighting. Constant current may also be important for some battery charging (see below). Constant current loads are rare in microgrids.

7.2.19.4 Battery Charging Output

A battery charger output port will implement a charging regime defined by the battery chemistry or the battery manufacturer, typically based on values and curves of the battery voltage, but perhaps by a Battery Management System (BMS).



(Note the horizontal axis here is ‘time’.)

Certain voltages and currents will be imposed for particular durations. Some of the time, a constant-voltage characteristic may be presented instead of constant-current. The values may also be a function of battery temperature.

The rate of charging may be determined by the limitations of available power.

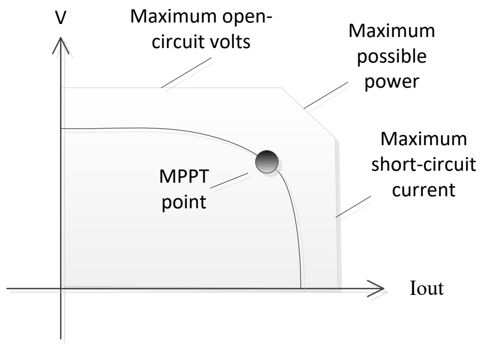
7.2.19.4.1 Schema

```
{
  "schema": {
    "$comment": "This schema is for a battery charging port",
    "type": "object",
    "properties": {
      "portType": {
        "type": "string",
        "const": "batteryCharging"
      },
      "nominalVoltage": {
        "anyOf": [
          { "$ref": "voltageRating-schema.json#/schema" },
          {
            "type": "array",
            "items": { "$ref": "voltageRating-schema.json#/schema" }
          }
        ]
      },
      "minNominalBattVolts": { "$ref": "voltageRating-schema.json#/schema" },
      "maxNominalBattVolts": { "$ref": "voltageRating-schema.json#/schema" },
      "chemistry": { "type": "string" },
      "BMScommunication": {
        "type": "object",
        "properties": {
          "protocol": {
            "anyOf": [
              { "type": "string" },
              {
                "type": "array",
                "items": { "type": "string" }
              }
            ]
```

```
              },
              "interface": {
                "anyOf": [
                  { "type": "string" },
                  {
                    "type": "array",
                    "items": { "type": "string" }
                  }
                ]
              }
            }
          }
        },
        "required": [ "nominalVoltage", "chemistry" ]
      }
```

7.2.19.5 Solar Input Port

File: `solarInputPort-schema.json`



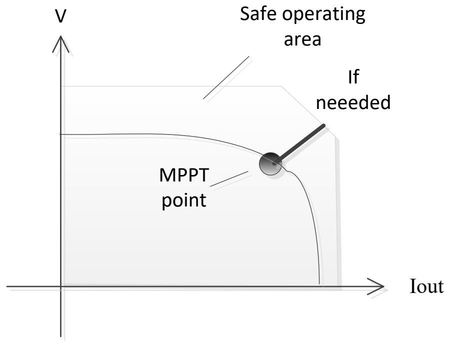
A solar input port will adjust the input conditions, typically using an MPPT algorithm. It will not have a fixed V-I characteristic, as this is determined by internal firmware to maximize the power extracted from the solar panel.

7.2.19.5.1 Schema

```
{
  "$comment": "This schema describes a solar or wind input port for connection
to solar panels.  In the case of wind/hydro, there is no requirement for the
input power to equal the output power.",
  "schema": {
    "type": "object",
    "properties": {
      "portType": {
        "type": "string",
        "const": "solarInput"
      },
      "SOA": { "$ref": "SOA-schema.json#/schema" },
      "maxOpenCctVolts": { "$ref": "voltageRating-schema.json#/schema" },
      "maxShortCctCurrent": { "$ref": "currentRating-schema.json#/schema" }
    }
  }
```

7.2.19.6 Wind/hydro input port

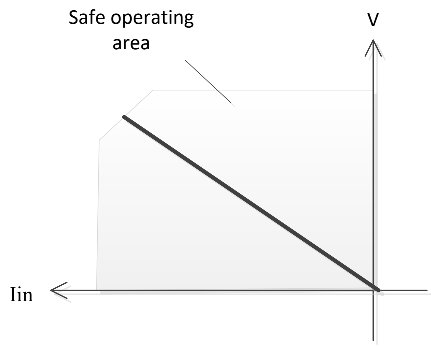
Small wind/hydro turbines may have an open interface, and use a third-party controller. For these, the solar panel schema may be used.



A wind/hydro input port will also attempt to maximize the power extracted, but if the power exceeds the permissible level, energy must be dissipated in a local load resistor, since disconnection would result in the rotor spinning out of control.

7.2.19.7 Resistive Load

File: `resistivePort-schema.json`



Both the oldest and probably the least important port type from a microgrid point of view!

7.2.20 General Port Schema

The following schema may be used for any port on any microgrid component:

7.2.20.1 Schema

```
"$comment": "This JSON schema is for any electrical power port",
"schema": {
  "type": "object",
  "properties": {
    "portName": { "type": "string" },
    "frequency": { "$ref": "frequency-schema.json#/schema" },
    "connectionType": { "$ref": "phase-schema.json#/schema" },
    "SOA": { "$ref": "SOA-schema.json#/schema" },
    "VIrelationship": {
      "oneOf": [
        {
          "type": "object",
          "properties": {
```

```
              "portType": { "const": "constantPower" },
              "powerLevel": { "$ref": "powerRating-schema.json#/schema" }
            }
          },
          {
            "type": "object",
            "properties": {
              "portType": { "const": "constantCurrent" },
              "currentLevel": { "$ref": "currentRating-schema.json#/schema" }
            }
          },
          {
            "type": "object",
            "properties": {
              "portType": { "const": "constantVoltage" },
              "openCctVolts": { "$ref": "voltageRating-schema.json#/schema" },
              "theveninResistance": { "$ref": "resistanceValue-schema.json#/schema" }
            }
          },
          {
            "type": "object",
            "properties": {
              "portType": { "const": "constantResistance" },
              "currentLevel": { "$ref": "resistanceValue-schema.json#/schema" }
            }
          },
          {
            "type": "object",
            "properties": {
              "portType": { "const": "batteryChargingPort" },
              "currentLevel": { "$ref": "batteryChargingPort-schema.json#/schema" }
            }
          },
          {
            "type": "object",
            "properties": {
              "portType": { "const": "VIrelationshipUndefined" }
            }
          }
        ]
      },
      "maxVoltageToGround": { "$ref": "voltageRating-schema.json#/schema" },
      "poleLocallyGrounded": { "$ref": "grounding-schema.json#/schema" },
      "groundWire": { "$ref": "PE-schema.json#/schema" },
      "connector": { "$ref": "connection-schema.json#/schema" },
      "prechargeProtection": {
        "type": "string",
        "enum": [ "provided", "required", "neither" ]
      },
      "crowbar": { "type": "boolean" }
    }
  }
}
```

7.2.21 Two-Port Passive Devices

A two-port passive device (TPPD) is a two-port component in which the output current and input current are equal, and the output voltage follows the input voltage. It does not contain a power source or energy storage.

A TPPD may have a single conductor (the current return path is implied), two power conductors carrying equal current, or exceptionally three or four conductors (eg 3-phase breakers).

Note: In the case of measuring and control TPPDs, there may be a third (data or control) port that does not carry significant power. This can still be a TPPD.

TPPDs include:

- Cables
- Switches, isolators, relay contacts
- Fuses and breakers
- kWh meters, ammeters

All share the following common characteristics:

- A maximum rated voltage (may be different for AC and DC)
    - For a switch/breaker/fuse, this will be the voltage when the connection between the port is open,
    - For a cable or other multi-pole device, this will be the maximum voltage between the poles or conductors
- A maximum rated voltage to ground (if not specified, deem to be the same as above)
- A maximum steady-state current
- A very small resistance between the two ports (in the case of two conductors, shared equally between them)
- Two connectors of some type, one for each port
- Product identification information, environmental constraints etc – parameters common to all components

7.2.21.1 Schema

```
"schema": {
  "type": "object",
  "properties": {
    "frequency": { "$ref": "frequency-schema.json#/schema" },
    "SOA": { "$ref": "SOA-schema.json#/schema" },
    "maxVoltageToGround": { "$ref": "voltageRating-schema.json#/schema" },
    "mechanicalContact": {"$ref":"mechanicalContact-schema.json#/schema"},
    "insertedResistance": { "$ref": "resistanceValue-schema.json#/schema" },
    "AEndConnection": { "$ref": "connection-schema.json#/schema" },
    "BEndConnection": { "$ref": "connection-schema.json#/schema" }
  }
}
```

Since the electrical characteristics of the ports are determined entirely by what they are connected to, only the port connectors are included in the schema.

For a TPPD, current is treated slightly differently, since it will always be the same at both terminals. Current from port A to port B is treated as positive, and in the reverse direction as negative. If only maxCurrentOut is specified, current rating is assumed to be in either direction ($I_{min} = -I_{max}$.)

8 Microgrid Component Data Structures

Each type of microgrid component has critical parameters that are specific to its function, in addition to the parameters common to all components.

8.1 Power Sources

8.1.1 Integrated Power Sources

File: `integratedSource-schema.json`

Any energy source that incorporates an integrated controller designed specifically for that energy source can be treated from an electrical point of view as a “black box” with a single output port, perhaps with the addition of one or more control ports. (This approach will not suffice when system functions such as energy balance are considered.)

A grid connection may be considered as an integrated power source (note that the port definition allows for bi-directional current flow).

8.1.1.1.1 Schema

```
{
  "$comment": "Any power source that includes a power controller",
  "schema": {
    "type": "object",
    "properties": {
      "outputPort": { "$ref": "port-schema.json#/schema" },
      "mechanical": { "$ref": "mechanical-schema.json#/schema" },
      "environmental": { "$ref": "environmental-schema.json#/schema" },
      "connector": { "type": "string" },
      "controlPort": { "$ref": "controlPort-schema.json#/schema" }
    }
  }
}
```

Where the power source and controller are from different manufacturers, each must be considered individually.

8.1.2 Solar Panels

Solar panels are the most popular renewable energy source for DC microgrids.

*8.1.2.1 Panel Type **

File: `panelType-schema.json`

8.1.2.1.1 Schema

```
"schema": {
      "type": "string"
  },
  "initialValues": [ "Building-integrated monofacial", "Flexible",
"Bifacial","Monofacial","Mono/bi-facial" ]
```

Building-integrated panels are panels designed to replace roof tiles/slates, rather than to be mounted above them (referred to as “building-applied”).

Monofacial panels only generate power from the sun shining on one side. Bifacial panels also generate some power from the back of the panel. This may give an uplift to the power output of

5-30%. Mono/bi-facial panels are intended to be used either with or without any sun shining on the back. Flexible panels can conform to modestly curved surfaces.

There will doubtless emerge other types of solar panel, which will need to be added to this list in due course.

8.1.2.2 Panel Technology

File: `panelTech-schema.json`

The microgrid designer may not be concerned with how the panel has been manufactured, as long as it does the job required.

8.1.2.2.1 Schema

```
    "schema": {
        "type": "string"
    },
    "initialValues": [ "Monocrystalline", "Polycrystalline", "Thin-film",
  "Perovskite" ]
  }
```

There are several material types used to manufacture solar panels. These are the important ones.

8.1.2.3 Electrical Characteristics

Since the electrical characteristics of a solar panel vary with temperature, they are typically quoted either as “STC” (Standard Temperature Conditions) or “NMOT” (Normal Module Operating Temperature) or “NOCT” (Normal Operating Cell Temperature). Although both the latter refer to an irradiance of 800 W/m^2, an ambient air temperature of 20°C, and a wind speed of 1 m/s, they are defined slightly differently, but each aspires to provide a more realistic performance in practice than STC (Standard Test Conditions, 25°C, irradiance of 1000 W/m^2).

For a “first-pass” selection process to select a better (or cheaper but equally good) product, it makes sense to compare like-for-like, for example to compare STC values for one with STC values for the other – even if neither product will actually meet these values in practice.

Regardless of the test conditions used, the performance figures quoted are:

8.1.2.3.1 Watts Peak *

The maximum power the panel is capable of generating (schema in §7.2.6). This is perhaps the most important parameter for a solar panel.

8.1.2.3.2 Open Circuit Volts *

The maximum voltage the panel can generate under no load (schema in §7.2.7). This is important for specifying the solar charge controller, as it will potentially have to withstand this voltage.

8.1.2.3.3 Short Circuit Current

The maximum current the panel can generate, when fed into a short circuit (schema in §7.2.5).

8.1.2.3.4 MPPT Volts and Current

The output voltage and current at the Maximum Power Point Tracking load conditions (when the peak wattage is being produced). This gives a more realistic value for the typical operating conditions when exposed to plenty of sunlight.

8.1.2.3.5 Efficiency

The percentage (0…100) of the incident radiation power that is converted to electricity. A figure of 20% is typical.

8.1.2.4 Bifacial Gain 5…30%

For bifacial panels, the electrical performance is enhanced by incident solar radiation on the back of the panel. This will normally be a fraction of the radiation hitting the front (a perfect mirror reflecting 100% of the sunlight hitting it to the back of the panel would give 100% bifacial gain, and double the power output).

8.1.2.5 Maximum System Voltage

Typically, several solar panels will be connected in series, raising the voltage to ground. The quality of the insulation around the panels determines the maximum voltage to ground that will be considered safe. The schema is defined in §7.2.7).

8.1.2.6 Maximum Fuse Rating

There will be a limit to the current the solar cells and cell interconnects can safely carry, regardless of any other factors. The string of panels should be fused by a fuse with a current rating no greater than this (schema in §7.2.5).

8.1.2.7 Integral Bypass Diode

Bypass diodes, also known as free-wheeling diodes, are wired within the PV module and provide an alternate current path when a cell or panel becomes shaded or faulty. They may or may not be included.

8.1.2.7.1 Schema

```
"integralBypassDiode": { "type": "boolean" },
```

8.1.2.8 Performance Warranty Years

Some manufacturers guarantee that their products will not degrade to more than a certain percentage within a certain number of years.

8.1.2.8.1 Schema

```
"performanceWarranty": {
  "years": { "type": "number" },
  "percentageReducedTo": {
    "type": "number",
    "minimum": 0,
    "maximum": 100
  }
}
```

*8.1.2.9 Mechanical Attributes **

Clearly, the size and shape of a solar panel is of critical importance – the number of panels is usually determined by the available area to mount them. However, the dimensions and weight can be defined in the same way as any other product. Therefore, this can use the schema defined in §7.2.2).

8.1.2.10 Environmental Parameters for Solar Panels

The environmental operating conditions (temperature, humidity, etc) that apply to any other product also apply to solar panels, and the definition in §7.2.4 may be used.

Environmental parameters specific to solar panels include the weight of snow per square meter they are guaranteed to survive, and the incident wind speed. These are not accounted for in this schema, as they will be unusual parameters to base product selection on.

8.1.2.11 Connector

All electrical components of a microgrid will have connections to other components via some kind of terminal or connector. The schema already caters for bolt terminations and screw clamp terminals, but solar panels typically are provided with MC3 or MC4-compatible single-pole connectors, with the female connector on the positive solar panel terminal (the polarity must be reversed for the connection to a solar charge controller). In this schema, a simple string defines the supplied connectors.

8.1.2.11.1 Initial dropdown list

Either “MC3” or “MC4”. Manufacturers must be empowered to add further connector types.

8.1.3 Rotating Energy Sources

This includes petrol and diesel generators, and wind and hydro turbines that do not have their own dedicated controller.

Petrol and diesel generators that include an integral inverter can be treated as integrated power sources with a single constant-voltage output port (see §8.1.1).

(more detail to follow in the next version of the IDM Standard)

8.1.4 Fuel Cells

8.2 Converters, Inverters and Power Supplies

A converter is a two-port component, without integral energy storage, and where one electrical characteristic (voltage, current or power) of one of the ports is controlled by hardware or software. It connects part of the system over which it has no control to part of the system that needs a parameter controlled.

These active microgrid components may include several integrated two-port passive devices (breakers, switches, fuses, etc) but the effect of these will be covered in the limitations of the overall product, and do not need to be documented separately.

Where the converter is unidirectional, it is normally referred to as a Power Supply if the output is DC, and as an Inverter if the output is AC.

With no energy storage capability, the hardware must ensure that the power input follows the power output with a certain efficiency, plus some static losses to power the internal hardware, the difference $(P_{in} - P_{out})$ being lost as heat:

$$P_{in} = \frac{P_{out}}{Efficiency} + P_{static}$$

At least one of the ports must have a constant-power characteristic, in order for the hardware to be able to balance the above equation. The converter will typically set whatever current is necessary to ensure that the required power flows.

If the converter is unable to balance the above equation, it must shut down, or compromise its port control regime so that it can.

The key parameters for a converter are therefore:

- What controls the amount of power that is converted – this may be defined by what the converter is connected to (the load or power source on the controlled port), or by external control or software.
- The voltage/current/power characteristics of the controlled port (the other will be constant-power)
- The safe voltage/current/power limits of the two ports,
- Whether each port is DC or AC, and if AC, how the frequency is determined.

A control port may also be present, which determines the behavior of the converter. This will have a physical interface, and a communications protocol.

8.2.1 Schema

```
  "$comment": "This JSON schema is for 2-port power converters and inverters,
but no energy storage",
  "schema": {
    "type": "object",
    "properties": {
      "port1": {
        "$ref": "port-schema.json#/schema"
      },
      "port2": {
        "$ref": "port-schema.json#/schema"
      },
      "staticPower": {
        "$ref": "powerRating-schema.json#/schema"
      },
      "transferPowerSetBy": {"type": "string", "enum":
["port1","port2","firmware","controlPort","the lower of P1 and P2"]}
      "controPort": { "$ref": "controlPort-schema.json#/schema" },
      "transferEfficicency%P1toP2": {
        "type": "number",
        "minimum": 0,
        "maximum": 100
      },
      "transferEfficicency%P2toP1": {
        "type": "number",
        "minimum": 0,
        "maximum": 100
      }
    }
```

8.2.2 Common microgrid converters

	Power flow set by	Controlled Port	Bi-directional?
AC-DC Power supply	DC load	Contant voltage DC (output)	No
DC-AC Inverter	AC load	Constant voltage AC (output)	No
AC or DC Battery charger	Battery algorithm	Battery charger output	No
Grid-tie inverter	Software	Constant power (both ports)	Maybe
LED Driver	Hardware	Constant current DC (output)	No

Solar charge controller	MPPT algorithm at low solar power, output voltage at high power	MPPT solar input (low power), Output port (high power)	No
Solar inverter	MPPT algorithm at low solar power, frequency at high power	MPPT solar input (low power), Output port (high power)	No
Solar battery charger	MPPT algorithm at low solar power, battery algorithm at high power	MPPT solar input at low solar power, battery charger at high power	No

8.2.3 Electric Vehicle Charging Points

An EV charging point (Electric Vehicle Supply Equipment, EVSE) is a special example of a two-port converter. Most commonly, EVSEs are unidirectional, charging the vehicle battery from the supply (in our case, a DC microgrid), but bi-directional products are slowly emerging that can use the vehicle battery to provide local storage for a small microgrid (this is usually referred to as "Vehicle-to-Home", V2H), and depending on the situation, it may also be used to provide local storage to support for the microgrid or grid as a whole (V2G). In all cases, internal firmware or software determines the behavior of the product.

8.2.3.1 EVSE Power levels

The power level of an EVSE can vary from 2.4kW up to 130kW or more, and this will be reflected in the time taken to charge the vehicle fully. These are referred to as:

- Level 1 (L1) – power sourced from a 120Vac socket. This will be power-limited by the circuit breaker, eg 20A (= 2.4kW)
- Level 2 (L2) – power sourced from a domestic 240V socket, or hard-wired into a domestic installation
- Leve 3 (L3) – DC fast charging

Naturally, an EV battery is DC, and in order to be able to charge your EV at home, the vehicle includes an AC-powered battery charger. However, this is power-limited to reduce weight and cost. Faster charging can be achieved by feeding DC directly to the battery, in which case the power is limited by the charging point and the amount of power it has access to. This is generally restricted to public charging points, and most of these are powered from the AC grid. However, there are obvious efficiencies to be achieved by using DC power if it is available, and of course IDM focuses on these.

8.2.3.2 EVSE Signaling Protocols

Standard protocols are emerging for communication between the charger and the vehicle – these include:

- Open Charging Point Protocol – several versions are in use (OCPP, IEC 63584) [1]
- IEC 63110

[1] See https://webstore.iec.ch/en/publication/95734

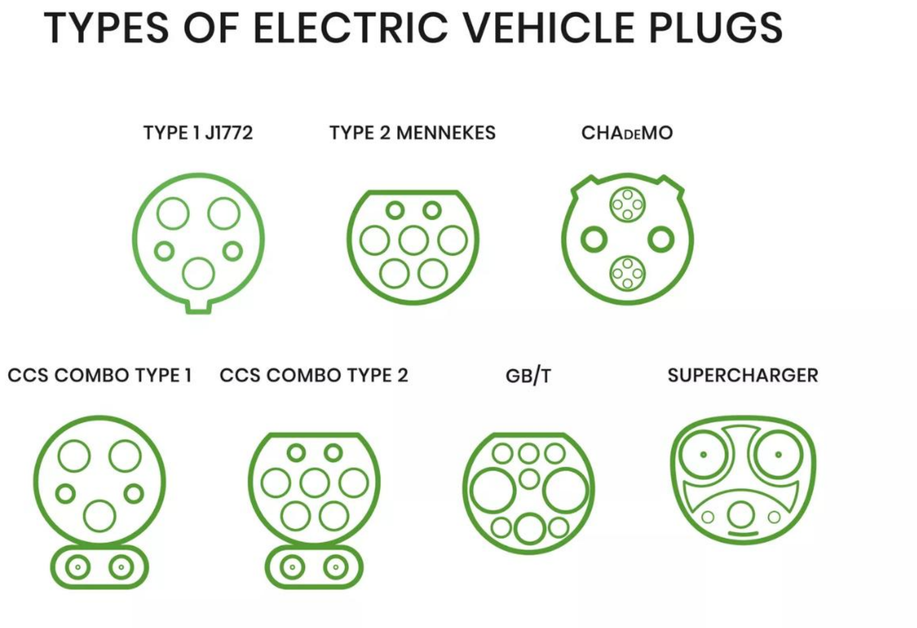
These EV – EVSE protocols are distinct from the protocols used for charging users for the use of public charging points, and protocols for managing the total demand from a parking lot with several charging points.

8.2.3.3 EV Charging Connectors

Specialized connectors are used for EV charging[2]. These include:

	L1 (AC)	L2 (AC)	L3 (DC)
J1772	√	√	
Mennekes	√	√	
Noth American Charging Standard, NACS (Tesla Supercharger)	√	√	√
Combined Charging System (CCS1 and CCS2) (backwards-compatible with J1772/Mennekes)	√	√	√
GB/T (China)	√	√	√
CHAdeMO / JEVS (usage declining)	√	√	√

A picture of these connectors may be helpful:



[3]

8.2.3.4 EVSE Control Ports

A typical EVSE will have one or more control port(s) supporting digital communication to allow real-time control of charging. The physical interface must be defined, and the communications protocol to be employed. The schema for a control port has been enhanced to reflect this.

[2] See https://www.lifewire.com/every-ev-charging-standard-and-connector-type-explained-5201160

[3] Permission for reproduction of this image has not been sought.

8.2.3.5 Including EV Charging in IDM

In order to include EVSEs in the IDM schema, the definition of a port must be extended to include the particular connectors and communication protocols employed in this situation. However, these are simply additional dropdown options within the existing schema.

The one exception is that EV chargers can deal with a wide range of nominal battery voltages. Two additional parameters are therefore added to the port definition: minNominalBattVolts and maxNominalBattVolts.

8.2.3.5.1 IDM EVSE Example

```
{
  "$schema": "common-schema.json#/schema",
  "IDMversion": "1.3 2025-08-10",
  "license": "(c) EMerge Alliance 2025, Licensed CC BY-NC-ND 4.0 -All Rights
Reserved",
  "productCategory":"converter",
  "productName": "60kW DC-DC Mobile EV Charger",
  "productIdentifier": "AMP-8002-60",
  "description": "This data is taken from https://dcide.app/products/60kw-dc-
dc-mobile-ev-charger-amp-8002-60-dc-dc-mobile-ev-charger-
676d68cdc8a1498c3e181318#electrical",
  "manufacturer": {
    "coName": "AmperneXt",
    "webHomePageURL": "www.ampernext.com"
  },
  "datasheetHyperlink": "https://www.ampernext.com/products/60kw-mobile-dc-dc-
ev-charger-dc-input/",
  "notRecommendedForNewDesigns": false,
  "typeSpecificParameters": {
    "componentType": "converter",
    "port1": {
      "portName": "input",
      "VIrelationship": {
          "portType":"constantPower",
          "powerLevel":"-60kW"
      },
      "frequency":"DC",
      "connectionType": "DC",
      "SOA":{
          "minVoltage":"300V",
          "maxVoltage": "820V",
          "maxCurrentIn": "-100A",
          "powerLimitIn": "-60kW"
      },
      "maxVoltageToGround": "1000V",
      "poleLocallyGrounded":"Float",
      "groundWire":"required",
      "connector": {
          "bolt": "M8 (5/16\")"
      },
      "prechargeProtection":"neither"
    },
    "port2": {
      "portName": "output",
      "VIrelationship": {
          "portType":"batteryCharging",
          "minNominalBattVolts":"150V",
          "maxNominalBattVolts":"1000V",
          "chemistry": "Li-ion",
          "BMScommunication":"Modbus TCP"
```

```
      },
      "frequency":"DC",
      "connectionType": "DC",
      "SOA":{
          "minVoltage":"150V",
          "maxVoltage": "1000V",
          "maxCurrentOut": "100A",
          "powerLimitOut": "60kW"
      },
      "maxVoltageToGround": "1000V",
      "poleLocallyGrounded":"Float",
      "groundWire":"offered",
      "connector": [
        {
          "type": "CCS1",
          "gender": "socket"
        },
        {
          "type": "CCS2",
          "gender": "socket"
        }
      ]
    },
    "controlPort": {
      "interface": [
        "10/100 Mbps Ethernet",
        "Wi-Fi",
        "3G/LTE"
      ],
      "controlProtocol": [ "OCPP 1.6j", "Modbus TCP" ]
    },
    "staticPower": "300W",
    "transferPowerSetBy": "controlPort",
    "transferEfficicency%P1toP2": 95
  },
  "environmental": {
    "ingressProtection_IP": "IP54",
    "DomesticComponentRequirement": false,
    "operatingTemperature": {
      "min": -25,
      "max": 55,
      "unit": "C"
    },
    "operatingHumidity%": {
      "max": 95
    }
  },

  "mechanical": {
    "dimensions": {
      "length": "870mm",
      "height": "480mm",
      "width": "670mm"
    },

    "weight": {
      "value": 100,
      "unit": "kg"
    }
  }
}
```

```
}
```

8.3 Energy Storage

8.3.1 Batteries

Batteries, flow batteries and fuel cells are used to provide energy storage for microgrids. As they represent a very significant fraction of the cost and space requirement of a microgrid installation, they are an area of very active technological development, and any schema must be ready to accept new technologies as they are made available. In this section, we focus solely on batteries, with a single bidirectional pair of terminals presenting the DC battery voltage (ie excluding products that include inverters, separate charging ports etc, but including batteries that have an integral battery management system (BMS) to ensure that all cells of the battery contribute equally.

*8.3.1.1 Nominal Voltage **

Batteries always have a quoted nominal voltage, which is usually somewhere near the middle of the typical voltage range of the battery. This is an essential first parameter when selecting a suitable product. The standard voltage rating definition in §7.2.7 may be used.

Clearly, for establishing electrical compatibility, the full possible voltage range will be important.

*8.3.1.2 Energy capacity **

The amount of energy the battery can store is also an important parameter. In practice, this is a function of:

- Battery temperature
- How fast the battery is charged and discharged
- How deep a discharge the user is willing to make the battery endure (almost all battery technologies suffer if the battery is discharged completely

In comprehensive battery documentation, curves will be supplied detailing how these parameters affect the energy stored.

In addition, the amount of energy put into the battery will be greater than the amount given out – the rest being dissipated as heat (or less desirably as permanent degradation of the internal chemistry). This is the so-called “round-trip efficiency”.

Despite all these caveats, it is essential that the manufacturer should provide an indication of the amount of energy the user should expect to get. The convention is that discharge capacity is quoted, at a certain discharge rate (eg C10 means discharging at a rate that discharges from full to empty in ten hours). A fast discharge (eg C1) will produce a much lower total energy capacity than a very slow discharge (eg C100). If a discharge rate is not quoted, C10 may be assumed.

The energy may be quoted in amp-hours (Ah), which can be multiplied by the nominal battery voltage to give the energy stored (watt-hours, Wh or kWh).

8.3.1.2.1 Schema

```
{
  "$comment": "This schema may be used for battery energy capacity",
  "schema": {
```

```
    "oneOf": [
      {
        "type": "object",
        "properties": {
          "value": { "type": "number" },
          "units": {
            "type": "string",
            "enum": [ "Ah", "Wh", "kWh" ]
          }
        },
        "required": [ "value", "units" ]
      },
      {
        "type": "string",
        "pattern": "^([0-9]+(.[0-9]+)?)(Ah|Wh|kWh)$"
      }
    ],
    "dischargeRate": {
      "type": "string",
      "pattern": "^C([0-9]+(.[0-9]+)?$"
    }
  }
}
```

*8.3.1.3 Chemistry **

There is a whole taxonomy of battery chemistries, and new ones are appearing almost daily. As each has its own strengths and weaknesses, selecting the optimal technology for a particular application becomes very important. Critical factors include:

- Safety issues (fire, outgassing, electrolyte spill, toxicity etc)
- Energy density (kWh/kg) – this will determine size and weight for a given energy capacity
- Cost (of course)
- Guaranteed number of discharge cycles (to a given discharge depth)
- Operating temperature range, and the impact of temperature on energy capacity

Once a preferred battery chemistry has been selected, it is essential that the electronics to charge the battery are configured to prevent overcharging, typically by setting the charging regime for the particular chemistry. It is also important for the control electronics to limit the discharge to the desired minimum charge level selected to optimize battery life against usable storage capacity.

For the purposes of this schema, the battery chemistry is simply a string, with a suggested dropdown list of initial values. The electronics associated with the battery should be selected to support the same technology (perhaps by means of manual configuration of voltages and charge times).

8.3.1.3.1 Suggested Initial Battery Chemistry Choices

Information taken from https://batteryuniversity.com/ .

- Lead-acid
  - Flooded (Wet)
  - VRLA (Valve-regulated Lead-Acid)
    - Standard, sealed
    - AGM (Absorbent Glass Mat)

        - Gel, carbon-gel
- Lithium (graphite anode)
    - Lithium Iron Phosphate ($LiFePO_4$)
    - Lithium Cobalt Oxide (LCO)
    - Lithium Manganese Oxide (LMO)
    - Lithium nickel manganese cobalt oxide (NMC)
    - Lithium nickel cobalt aluminum oxide (NCA)
- Lithium Titanate anode
    - Lithium nickel manganese cobalt oxide (NMC)
    - Lithium nickel cobalt aluminum oxide (NCA)
- Nickel Metal Hydride (NiMH)
- Nickel-Cadmium
- Sodium-ion

8.3.1.4 Battery Terminals

Many companies have used the same definitions for battery terminals, viz:

> **Auto Post Terminal (SAE terminal)**
>
> This is the most common battery terminal type, and any person who has replaced a car battery can easily recognize it. In order to prevent accidently connecting the terminals in reverse polarity, the positive post is always larger diameter than the negative. Another terminal that you will find is what is known as Pencil Post (found predominantly in batteries for Japanese cars – JIS types). When compared with a SAE terminal, the Pencil Post is smaller.
>
> **Stud Terminal**
>
> This is a 3/8" threaded stainless steel terminal is designed to fasten and hold the terminal connection to the terminal lug onto the lead base of the terminal.
>
> **Dual Post Terminal / Marine Terminal**
>
> This terminal type has an Automotive Post and a Stud (5/16"). You can make the connection using either a traditional pressure contact or a ring terminal and wing nut connection.
>
> **Button Terminal**
>
> These are also known as insert terminals. You will find these terminals from M5 to M8 which refers to the metric size of the diameter of the bolt thread. For example, if you have a battery with a M8 terminal, you will need a bolt with an 8 millimetre diameter thread. These types of terminals are most commonly found on Absorbed Glass Mat batteries used in emergency backup and uninterruptable power systems (UPS) battery applications.
>
> **AT Terminal (Dual SAE / Stud type terminals)**
>
> They are commonly found in traction type batteries used in heavy cycling applications such as floor scrubbers and off-grid solar application batteries. This terminal type has an Automotive Post and a Stud (3/8" threaded stainless steel terminal).

I have therefore added an “other” option to the connection-schema (see §7.2.177.2.17.17.2.16), with the only suggested dropdown value as “SAEterminal”. (Stud and Button can both use the “bolt” value.)

8.3.1.5 Battery Management System

Certain battery chemistries have the characteristic that cells connected in series may not balance automatically (certain cells taking more charge than others), leading to some cells being overcharged while others are undercharged. Battery management systems (BMS) exist to correct this, and to control the overall amount and rate of charge. This is essential to prevent batteries overheating, with potentially disastrous consequences.

A BMS will communicate with the battery charger (and potentially discharger) to ensure that charging is managed correctly. The communications interface and protocol need to be specified.

8.3.1.5.1 Schema

```
"BMScommunication": {
  "type": "object",
  "properties": {
    "protocol": {
      "anyOf": [
        { "type": "string" },
        {
          "type": "array",
          "items": { "type": "string" }
        }
      ]
    }
  },
  "interface": {
    "anyOf": [
      { "$ref": "string" },
      {
        "type": "array",
        "items": { "type": "string" }
      }
    ]
  }
}
```

Note that even if the interface and protocol data check out, there will remain many fine details of the communication protocol that could give rise to incompatibility between the charge/discharge controller and the battery.

8.3.2 Flywheel Storage

(to follow)

8.3.3 Supercapacitors

(to follow)

8.3.4 Thermoelectric Storage

(to follow)

8.3.5 Gravity-based Storage

(to follow)

8.4 Loads

File: `load-schema.json`

IDM does not concern itself with the function the load performs, but only with the electrical and limited mechanical parameters associated with it.

8.4.1 Load Schema

```
{
  "$comment": "This JSON schema is for any electrical load",
  "schema": {
    "type": "object",
    "properties": {
      "powerInput": {
        "$ref": "port-schema.json#/schema"
      },
      "controPort": { "$ref": "controlPort-schema.json#/schema" }
    },
    "required": ["powerInput"]
  }
```

Load inductance and capacitance are for a future version of the IDM Standard that includes dynamic properties.

8.4.2 Motor Loads

(to follow)

8.4.2.1 AC Induction Motors

8.4.2.2 DC Commutator Motors

8.4.2.3 Brushless DC Motors

These may be treated as electronic loads (see §8.4.5).

8.4.3 Lighting Loads

Lighting fittings that incorporate control circuitry may be treated as electronic loads (see §8.4.5). Some LED lamps without control circuitry require a constant-current driver.

8.4.4 Heating Loads

These will usually have a constant-resistance V-I characteristic.

8.4.5 Electronic Equipment

These will usually have a constant-power V-I characteristic.

8.5 Power Distribution

These are passive devices, where the current through the two ports is the same, and the voltage is almost the same (unless the component opens the circuit).

Passive two-port devices share many common parameter definitions, which are captured in the Two Port Passive Device (TPPD) schema in §7.2.18 above.

8.5.1 Cables

File: `cable-schema.json`

Cables are a simple example of a TPPD. In this case, the insertedResistance parameter is the resistance per meter of the cable. (Although it is obviously possible to have cables with conductors of different sizes, insulation colors, screenings etc, this schema does not support that level of detail.)

8.5.1.1 Schema

```
"schema": {
  "type": "object",
  "properties": {
    "TPPD": {"$ref": "TPPD-schema.json#/schema"},
    "numberOfConductors": {
      "type": "integer",
      "minimum": 1
    },
    "length": {"$ref": "dimensions-schema.json#/schema"},
    "conductorSize": {"$ref": "wireSizes-schema.json#/schema"},
    "strandsPerConductor": {
      "type": "integer",
      "minimum": 1
    },
    "conductorMaterial": {
      "type": "string",
      "enum": [
        "copper",
        "aluminum",
        "AAC",
        "AAAC",
        "ACSR"
      ]
    },
    "insulationMaterial": {"type": "string"},
    "armored": {"type": "boolean"},
    "environmental": {"$ref": "environmental-schema.json#/schema"}
  }
}
```

8.5.2 Switches and isolators

File: `isolator-schema.json`

Switches and isolators are important elements of a microgrid system.

Note: DC Isolators should be exercised periodically, to wipe the contacts and ensure that contact resistance remains low.

8.5.2.1 Schema

```
"schema": {
  "type": "object",
  "properties": {
    "rating": {"$ref": "mechanicalContact-schema.json#/schema"},
    "numberOfPoles": {
      "type": "integer",
      "minimum": 1
    },
    "mechanical": {"$ref": "mechanical-schema.json#/schema"},
```

```
              "lockable": {"type": "boolean"},
              "style": {
                "type": "string",
                "enum": [
                  "On/Off",
                  "A-Off-B",
                  "Changeover"
                ]
              }
            }
          }
```

8.5.3 Relays and Contactors

This structure does not consider speed of operation. This may be important in some circumstances.

8.5.3.1 Mechanical Relays

File: `contactor-schema.json`

Contactors and relays have mechanical contacts controlled by a coil.

, or the equivalent function implemented in semiconductors with opto-isolation.

(Solid-state relays are to follow)

8.5.3.1.1 Schema

```
          "schema": {
            "type": "object",
            "properties": {
              "coilNominalVolts": {"$ref": "voltageRating-schema.json#/schema"},
              "coilACDC": {
                "type": "string",
                "enum": [
                  "AC",
                  "DC",
                  "AC/DC"
                ]
              },
              "coilResistance": {"$ref": "resistanceValue-schema.json#/schema"},
              "coilSOA": {"$ref": "SOA-schema.json#/schema"},
              "contactRating": {"$ref": "TPPD-schema.json#/schema"},
              "coilContactIsolationVolts": {"$ref": "voltageRating-schema.json#/schema"},
              "numberOfPolesNormallyOpen": {
                "type": "integer",
                "minimum": 0
              },
              "numberOfPolesNormallyClosed": {
                "type": "integer",
                "minimum": 0
              },
              "numberOfPolesChangeover": {
                "type": "integer",
                "minimum": 0
              },
              "numberOfPolesMakeBeforeBreak": {
                "type": "integer",
                "minimum": 0
              },
              "mechanical": {"$ref": "mechanical-schema.json#/schema"},
```

```
                "connection": {"$ref": "connection-schema.json#/schema"}
              }
            }
```

8.5.3.2 Solid State Relays

Solid-state relays provide the equivalent function implemented in semiconductors with opto-isolation.

(Solid-state relays are to follow)

8.5.4 Energy Meters

A kWh meter has no special electrical requirements, beyond those of any other TPPD. The TPPD Schema may be used.

8.5.5 Busbars

(to follow)

8.5.6 Distribution Boards

(to follow)

8.6 Circuit Protection

8.6.1 Fuses

File: `fuse-schema.json`

In this case, the assumption is that a fuse is a replaceable component, and therefore must be accessible. The physical shape is important, but obviously the current rating is the critical factor from an electrical point of view.

8.6.1.1 Schema

```
{
  "schema": {
    "type": "object",
    "properties": {

      "TPPD": { "$ref": "TPPD-schema.json#/schema" },
      "fuseType": { "$ref": "fuseType-schema.json#/schema" },
      "breakingCapacity": { "$ref": "currentRating-schema.json#/schema" },
      "responseTime": { "$ref": "fuseResponse-schema.json#/schema" },
      "blownIndicator": { "type": "boolean" }
    }
  }
}
```

8.6.1.2 Fuse Shape and Physical Size *

File: `fuseType-schema.json`

Fuses come in many shapes and sizes. For microgrids, the most popular formats are “Cartridge”, “Flush square body”, “Blade”, “L25S/L50S” and “SQB”, and the dropdown list will initially be populated with these – but inevitably, others will need adding. (We do not consider fuses that are soldered in place.) Automotive blade fuses are popular for low-power 12V and 24Vdc microgrids. Each format has a range of sizes – but each format uses its own terminology.

*8.6.1.3 Current Rating **

File: `currentRating-schema.json`

The maximum continuous load current the fuse will pass indefinitely without blowing.

The format is defined in §7.2.5.

8.6.1.4 Maximum Breaking Current

File: `currentRating-schema.json`

The maximum fault current the fuse will interrupt.

The format is defined in §7.2.5.

8.6.1.5 Maximum Breaking Voltage (AC/DC)

File: `voltageRating-schema.json`

The maximum voltage across the fuse terminals after the fuse has blown. As there is a real possibility of an arc between the ends of the broken fuse element, this voltage is always equal or less for DC than for AC.

The format is defined in §7.2.7.

8.6.1.6 Fuse Speed of Response

File: `fuseResponse-schema.json`

Fuses allow a certain amount of energy to pass in excess of the rated current before they blow. Slow-blow fuses tolerate more than fast-blow fuses, which in turn tolerate more than fuses designed to protect semiconductors.

8.6.1.6.1 Initial schema values

```
    "Semiconductor",
    "Fast blow (F)",
    "Normal (M)",
    "Slow blow (T)",
    "Time delay (TT)"
```

Manufacturers may add to this list. For further information, see IEC60269, or https://www.swe-check.com.au/pages/learn_fuse_speed.php. A more quantitative treatment would use the I^2T characteristic curves supplied by the manufacturer.

8.6.1.7 Blown Fuse Indicator

Some fuses are provided with an indicator that changes color or appearance when the fuse blows. This is just a Boolean parameter indicating whether such functionality is present.

8.6.1.7.1 JSON Schema

```
{
  "schema": {
    "type": "boolean”
  }
}
```

8.6.2 Breakers

File: `breaker-schema.json`

A breaker opens a circuit if excessive current flows. It may also function as a manual on/off switch. After an overcurrent has occurred, most breakers require a manual reset, but a few are “reclosers”, closing again automatically two or three times in case the fault has cleared itself, or are resettable remotely.

8.6.2.1 Schema

```
    "schema": {
      "type": "object",
      "properties": {
        "TPPD": {"$ref": "TPPD-schema.json#/schema"},
        "numberOfPoles": {
          "type": "integer",
          "minimum": 1
        },
        "tripCriteria": {"$ref": "tripCriteria-schema.json#/schema"},
        "detectionMethod": {"$ref": "breakerType-schema.json#/schema"},
        "unsafeIfDCpresent": {"type": "boolean"},
        "isolationMechanism": {
          "type": "string",
          "enum": [
            "mechanical",
            "solid-state",
            "hybrid"
          ]
        },
        "contactRating": {"$ref": "mechanicalContact-schema.json#/schema"},
        "voltageRatingDC-2PolesInSeries": {"$ref": "voltageRating-
  schema.json#/schema"},
        "tripCurve": {"$ref": "breakerTripCurve-schema.json#/schema"},
        "moechanical": {"$ref": "mechanical-schema.json#/schema"},
        "connection": {"$ref": "connection-schema.json#/schema"},
        "reset": {
          "type": "string",
          "enum": [
            "manual",
            "auto",
            "remote"
          ]
        },
        "auxiliaryContact": {
          "type": "string",
          "enum": [
            "NC",
            "NO",
            "C/O"
          ]
        }
      },
      "required": ["currentRating"]
   }
```

8.6.2.2 Trip Criteria *

Breakers are designed to interrupt the current if an anomalous situation occurs. This may be:

- Overcurrent
- Under – or Over-Voltage
- Leakage to ground (“ground fault”)
- Arcing

- Phase imbalance
- Manual turn-off by a user

Some breakers can also be tripped by an external solenoid.

The schema allows for each of these trip mechanisms to be specified in any combination. However, breakers are often referred to by acronyms according to their trim mechanism(s) – though note that this usage is not always consistent, and the terminology in Europe differs from that in the US:

	GFCI	AFCI	RCB	RCD	RCBO	RCCB	MCB	MCCB	AFCB	ELCB
Overcurrent					Y		Y	Y		
Over/undervoltage										
Ground leakage	Y		Y	Y	Y	Y				Y
Arcing		Y							Y	

Therefore, the schema permits one of the above acronyms in lieu of specifying the criteria individually.

8.6.2.2.1 JSON Schema

```
{
  "$comment": "Reasons that a breaker might turn off.",
  "schema": {
    "type": "object",
    "parameters": {
      "overCurrent": { "type": "boolean" },
      "overVoltage": { "type": "boolean" },
      "underVoltage": { "type": "boolean" },
      "groundFault": { "type": "boolean" },
      "arcFault": { "type": "boolean" },
      "manualOperation": { "type": "boolean" },
      "phaseImbalance": { "type": "boolean" },
      "externalSolenoid": { "type": "boolean" },
      "acronym": {
        "type": "string",
        "enum": [
          "GFCI",
          "AFCI",
          "RCB",
          "RCD",
          "RCBO",
          "RCCB",
          "MCB",
          "MCCB",
          "AFCB",
          "ELCB"
        ]
      }
    }
  }
}
```

8.6.2.3 Detection Technology

Several detection methods are in common use, each with particular strengths and weaknesses.

The two principal methods for detecting overcurrent are magnetic (the current in a coil attracts an armature) or thermal (a bimetallic strip heats up) and in either case, the resulting movement releases a catch. The coil acts fast, the bimetallic strip is slower. Where the permissible surge current duration is several seconds, an additional hydraulic damper may slow the process further.

Some breakers employ electronics to detect anomalous conditions (this is always true for arc fault detection, and over/undervoltage detection).

8.6.2.3.1 Dropdown List of Detection Methods

- "Thermal",
- "Thermal-magnetic",
- "Magnetic",
- "Thermal-magnetic-hydraulic",
- "Electronic",
- "Hybrid"

This list is probably complete until some new technology is developed.

*8.6.2.4 Number of Poles **

A breaker can interrupt a number of current-carrying conductors simultaneously (for example, a three-phase breaker may interrupt the three live conductors, or those and the neutral).

8.6.2.4.1 Schema

```
{
  "schema": {
    "type": "object",
    "properties": {
      "numberOfPoles": {
        "type": "integer",
        "minimum": 1
      },
```

*8.6.2.5 Current Rating **

The maximum steady-state current the breaker will allow. As tripping follows a curve gradually reducing the time taken to trip as the current over the rated current increases, a current marginally over the rated current *could* trip the breaker, but it might take a very long time.

The format for the current is defined in §7.2.5.

8.6.2.6 Voltage Rating AC/DC

When the breaker is closed, the voltage across the terminals is minimal, but when the breaker contacts open, the full supply voltage is presented across them. If the contacts are mechanical, there will be some arcing, which will be short-lived if the supply is AC, but could continue indefinitely with DC. Therefore the voltage rating for DC will always be lower than for AC. Some manufacturers specify an increased DC voltage by connecting two opening poles in series to double the arc length.

8.6.2.6.1 Voltage Rating Schema

```
"voltageRatingAC": { "$ref": "voltageRating-schema.json#/schema" },
"voltageRatingDC": { "$ref": "voltageRating-schema.json#/schema" },
"voltageRatingDC-2PolesInSeries": {"$ref": "voltageRating-schema.json#/schema"},
```

The format for the voltage is defined in §7.2.7.

8.6.2.7 Breaking Capacity

When a short-circuit occurs, the current may initially be very large – many times the maximum current the breaker is designed to allow to pass. This parameter specifies the maximum breaking current the breaker can interrupt.

The format for the current is defined in §7.2.5.

8.6.2.8 Overcurrent Trip Curves

There are many loads that require a high current briefly when first powered up – for example, large motors. Breakers are chosen according to the degree of overcurrent and its duration that are required not to trip the breaker.

8.6.2.8.1 Initial Suggested dropdown list of trip curves

- "IEC 60947-2 Type Z",
- "IEC 60898-1 Type B",
- "IEC 60898-1 Type C",
- "IEC 60947-2 Type K",
- "IEC 60898-1 Type D",
- "IEC 60947-2 Type MA",
- "IEC 60934",
- "Custom"

8.6.2.9 Isolation Mechanism

Breakers can interrupt the current either by opening mechanical contacts, or by turning off solid-state semiconductors, or by a combination of the two.

8.6.2.9.1 JSON Schema

```
"isolationMechanism": {
  "type": "string",
  "enum": [ "mechanical", "solid-state", "hybrid" ]
},
```

This is probably a complete list until some new technology arrives.

8.6.2.10 Mounting Style

This is defined in §7.2.13.

8.6.2.11 Connections

This is defined in §7.2.16.

8.6.2.12 Reset Mechanism

After a breaker has tripped, it must be reset to restore the connection. There are really only three options:

- Manual (the default if not specified)
- Auto – this mostly applies to reclosers for high voltage systems, which will try to restore the current two or three times before giving up
- Remote – an external command sent via some communications method

8.6.2.12.1 Schema

```
"reset": {
  "type": "string",
  "enum": [ "manual", "auto", "remote" ]
```

8.6.2.13 Auxiliary Contact

Some breakers have an auxiliary contact, which may be used to trigger an alarm, or ensure that some other equipment does not remain powered after the breaker has tripped. Normally-closed (NC), Normally-open (NO) and Changeover (C/O) auxiliary contacts are possible.

8.6.2.13.1 Schema

```
"auxiliaryContact": {
  "type": "string",
  "enum": [ "NC", "NO", "C/O" ]
```

8.7 Multi-port products

Microgrid components that have integrated energy storage, and/or more than two ports, have internal processes that are too complex and varied to be easily modelled. Consequently, their descriptive schema is relatively simple:

8.7.1 General Schema

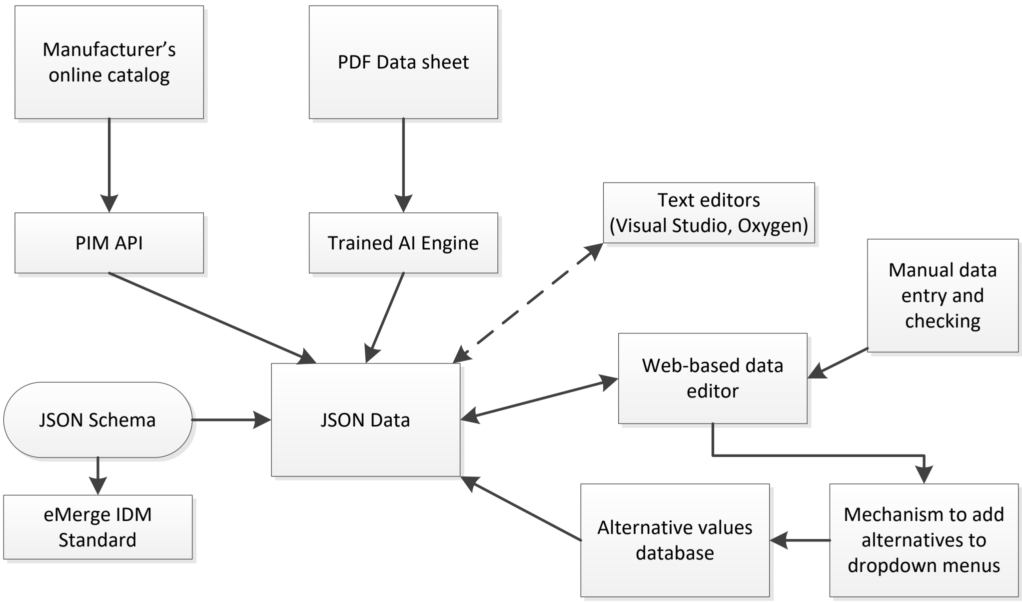
```
"schema": {
  "type": "object",
  "properties": {
    "ports": {
      "type": "array",
      "items": {"$ref": "port-schema.json#/schema"}
    },
    "controPort": {"$ref": "controlPort-schema.json#/schema"},
    "functions": {
      "type": "array",
      "items":{
        "type":"object",
        "properties": {
          "fromPort": { "type": "string" },
          "toPort": { "type": "string" },
          "function": {
            "type": "string",
            "enum": [
              "solar battery charge controller",
              "solar to AC grid-tie inverter",
              "grid pass-thru",
              "AC-powered battery charger",
              "direct solar to load",
              "AC-DC power supply",
              "DC to AC inverter (non-grid-tie)",
              "DC to AC inverter (grid-tie)",
              "bidirectional DC-AC grid tie inverter",
              "battery discharge controller"
            ]
          },
          "ratedPower": { "$ref": "powerRating-schema.json#/schema" },
          "surgeCapability": {
            "type": "object",
            "properties": {
```

```
                    "power": { "$ref": "powerRating-schema.json#/schema" },
                    "duration": {"$ref": "duration-schema.json#/schema" }
                  }
                },
                "transferEfficiency": {
                  "type": "number",
                  "exclusiveMinimum": 0,
                  "exclusiveMaximum": 100
                }
              }
            }
          },
          "outputsParallelable": {"type": "boolean"},
          "3for3phase": {"type": "boolean"},
          "staticPower": {"$ref": "powerRating-schema.json#/schema"},
          "energyCapacity": {"$ref": "energy-schema.json#/schema"}
        }
      }
```

9 Populating the Dataset

A data structure is only as useful as the data that is created using it. The time taken to manually enter data can quickly become significant, and a degree of automation is highly desirable.

The process is envisioned to be like this:



9.1 Text Editors

Text editors that incorporate JSON syntax checking (eg Microsoft Visual Studio, Syncro Soft Oxygen XML) may be used to manually enter product data, but this is quite slow and prone to errors.

9.2 Creation of Product IDM Files using AI

It is intended that creation of the JSON file describing a product using AI is supported. Publicly available AI engines (eg ChatGPT) do a reasonable job of answering product-related questions, *provided that a PDF datasheet has been uploaded*. (Without an uploaded datasheet, they are liable to make up the answers!)

To achieve this, a prose question is offered, and a Regular Expression used to extract the parameter value from the returned verbose reply. *[This is not implemented in the current version.]*

9.3 Creation of Product IDM Files by Linking with Manufacturers' Catalogs

There are several companies that offer Product Information Management (PIM) systems. These are widely adopted by the larger manufacturers of microgrid components. However, each PIM system has its own unique Application Programming Interface (API). In order to create connectors between manufacturers' systems and IDM, substantial work is likely to be required. However, the reward (automatic inclusion of all the manufacturer's products in IDM) is considerable.

10 Using the Dataset

10.1 Selection Tool

This data structure has been designed with the objective of making it easy to select a suitable product from the IDM Catalog, given a list of required parameters. The process is typically:

- Select the type of microgrid component sought
- Enter a value or values for several relevant parameters
- Filter the available products, and review the list
- Select a small number of promising candidates
- Compare these side-by-side, and make a selection
- (If required) download detailed data on the selected product(s) from the manufacturer

There will therefore be three panes on web pages:

- A **Filter** area (usually at the top or left-hand side) with columns for parameter values to filter on, with the possibility of selecting several text values, or a min/max range for numerical parameters,
- A **Display** area, which may show some or all of the filtered products, one line for each (usually below the filter pane),
- A **Compare** page, where a small number of selected products are listed in columns, with the same parameters in the same order, to facilitate seeing the differences between them.

Some of the JSON Schemas include parameters to specify how the three selection tool areas of web pages might be presented. An example of this is for a battery might be:

```
          "selectionTool": {
            "filter": [ "chemistry", "nominalVoltage", "energyCapacity" ],
            "display": [ "image","chemistry", "nominalVoltage",
        "energyCapacity","listPrice" ],
            "compare": ["chemistry", "nominalVoltage",
        "energyCapacity","length","width","height","weight","listPrice" ]
          }
```

10.2 Checking Compatibility

When checking electrical compatibility, only limiting (max/min permissible) values are considered – typical values are ignored.

There are four stages to determining whether the ports on two electrical components can be connected together, without causing equipment misbehaviour or permanent damage:

1. Are the ports compatible from the point of view of polarity (DC) or frequency and phase (AC) of what is offered with what is expected? (§10.2.1
2. Do the Safe Operating Areas (SOA) of the two ports overlap? (§10.2.2)
3. If it is possible to anticipate the voltage and current on the connection, will the values fall within the overlapping SOA? (§10.2.3)
4. Does only one of the ports have a direct connection to ground (DC or AC) or if both have a connection to ground (AC only), is the same conductor grounded in both? (Multiple DC ground connections are only permitted if at least one of the connections is via enough diodes for their forward conducting voltage to exceed any possible ohmic voltage difference.) (§10.2.4)

(This only determines whether connecting these component ports is likely to cause damage. It doesn't say whether it makes sense from a functional point of view.)

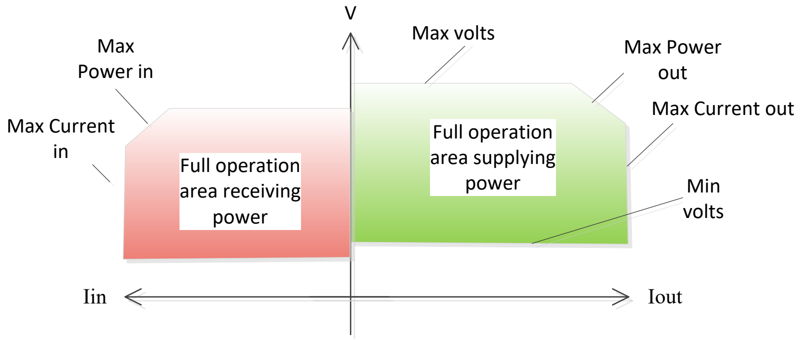
10.2.1 Checking Frequency

The Summing section of the Frequency schema (`frequency-schema.json`) lists the permissible combinations of the Frequency parameter, and the net result in the event a third component should be connected in parallel.

10.2.2 Checking Safe Operating Area

Every port has limits to the voltage, current and power that may be transmitted through it, and also a limit to the voltage to ground that may be present on the port while continuing to be safe. The voltage, current and power limits define a Safe Operating Area (SOA). If two ports are connected together, their SOAs (with the sign of the current changed on one) must overlap. The new overlapping SOA is the operational limit when these two ports are connected together. This provides a first-pass test for whether the two ports may be connected together. If there is no overlapping area, the ports must not be connected.

Graphically, this may be represented as follows:



10.2.2.1.1 Terminal Voltage Limits

When two ports A and B are connected together, a new (smaller) SOA is created:

$$V_{max} = \min\left(V_{Amax}, V_{Bmax}\right)$$

$$V_{min} = \max\left(V_{Amin}, V_{Bmin}\right)$$

For the two ports to be compatible, it is required that $V_{max} > V_{min}$ and $I_{limit} > 0$. If a third port is connected in parallel, these new values must be used to assess the compatibility of the third port.

If any of these values are unavailable, default values for V_{max}, V_{gnd} should be set to impossibly high values and V_{min} should be set to zero – though of course there is a risk to doing this.

10.2.2.1.2 Terminal Current Limits *

It is assumed that zero current will never damage equipment, but equally nothing will function. The limiting factor will always be the maximum current the port can withstand, in one direction or the other. Specifically: I_{max} is the maximum current that the port can output (ie source), I_{min} is the maximum current that the port can input (ie sink).

Note: Limiting current values cannot be defaulted, and must be known.

For example, a power supply will typically have I_{max} set to some value, and I_{min} is zero. A load will have I_{max} set to zero, and I_{min} at the (negative) maximum current the load will demand when it is operating. It is always acceptable to have more current available than is required, thus the compatibility criterion is:

$$\sum I_{\max[n]} + \sum I_{\min[n]} \geq 0$$

(That is to say, there must be current to spare.) Note that when several loads are connected in parallel, and diversity is taken into account, and provided the possibility of overcurrent protection tripping is acceptable, then a much lower powered source may be satisfactory.

10.2.3 Predicting the Terminal Voltage and Current

In addition to SOA compatibility, in some cases it is possible to determine the voltage and current that will arise when two microgrid component ports are connected together. This is only possible where the voltage-current relationship for both ports is defined. (In many applications, the desired power flow is determined by software, rather than – or in addition to – hardware or the electrical conditions on the ports.)

Discussion point: Where a range is specified for the V-I relationship, should the worst case be used? How will the worst case be identified?

When two unipolar DC or single-phase AC ports are to be connected together, and the V-I characteristics of both are known, it is possible to calculate the voltage and current that will result. Clearly, it is important that the resulting voltage, current and power values are within the SOA of both ports.

There is equipment where a port does not have a defined V-I characteristic – an example is the solar input of a solar charge controller. This check is not possible in this case, and this is also the situation for any passive device, unless its internal resistance is added to the V-I characteristics of whatever it is connected to.

The formulae describing the likely V-I curves are:

Constant voltage output	$V = V_{oc} - I_{out}R_{thevenin}$	$I_{out} = (V_{oc} - V)/R_{thevenin}$
Constant voltage input	$V = V_{oc} + I_{in}R_{thevenin}$	$I_{in} = (V - V_{oc})/R_{thevenin}$
Constant current		$I = I_k$
Constant power out	$V = P\ /\ I_{out}$	$I_{out} = P\ /V$
Constant power in	$V = P\ /\ I_{in}$	$I_{in} = P\ /V$
Constant resistance load	$V = I_{in}R$	$I_{in} = V/R$

To solve, substitute the right-hand formula for I_{in} for one port for I_{out} in the left-hand equation for the port to which it is to be connected (sometimes, it's easier to do it the other way round).

10.2.3.1 Formulae for constant-voltage sources

A constant-voltage output driving a resistive load has the formula:

$$V = V_{oc} - R_{thevenin}V/R)$$

This gives:

$$V = V_{oc}/(1 + \left(\frac{R_{thevenin}}{R}\right)) \quad \text{and of course} \quad I \quad = V/R$$

The same exercise for constant-voltage driving a constant-current load is trivial:

$$V = V_{oc} \quad - I_k R_{thevenin}$$

For a constant-power load,

$$V = V_{oc} - R_{thevenin}\frac{P}{V}$$

$$V^2 = VV_{oc} - R_{thevenin}P$$

$$V^2 - VV_{oc} + R_{thevenin}P = 0$$

Since $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$, $a = 1,\ b = -V_{oc}\ , c = R_{thevenin}P$, the solution will be

$$V = \frac{V_{oc} \pm \sqrt{{V_{oc}}^2 - 4R_{thevenin}P}}{2}$$

Putting some realistic numbers into this, $V_{oc} = 12V,\ R_{thevenin} = 0.1\Omega, P = 24W$, then

$$V = \frac{12 \quad \pm \sqrt{144 - 9.6}}{2}$$

The terminal voltage is (12 ± 11.59)/2 = 11.796V. Substituting for the current gives 2.0067A

(It could actually also be 0.205V – but that would imply a current of 117amps, which would be bad news for the power supply! It would also obviously breach both SOAs. This will generally be the case for one of the two solutions.)

10.2.3.2 Other Combinations

Using droop control, two constant-voltage ports are permitted to be connected, and a current will flow out of one and into the other. If current flowing from A to B is positive and B to A negative, the applicable formula is:

$$I = (V_{oc(a)} - V_{oc(b)})/(R_{droop(a)} + R_{droop(b)})$$

$$V = V_{oc(a)} - (\frac{(V_{oc(a)} - V_{oc(b)})}{(R_{droop(a)} + R_{droop(b)})})R_{droop(a)}$$

This formula may also be used for a constant-voltage battery charger charging a battery (where the battery chemistry permits that).

In addition, these combinations are theoretically possible, but unlikely:

- A constant-current source and a constant-power or resistive load
- A constant-current source and a constant-voltage load (eg an LED lamp) (simply $I = I_k, V = V_{led}$ – both will automatically be within the SOA)
- A constant-current load with a constant-power source
- A renewable energy source connected directly to any type of load (what happens will depend on the availability of power)

For these combinations, just check SOA limits to determine compatibility.

Other possible port combinations are not permitted. Constant-power source to constant-power sink would be unable to agree the power level. Likewise constant-current to constant-current.

10.2.4 Voltage to Ground

If there is a limit to the safe voltage (of any power pole) to ground, the new value must be the more restrictive of the two ports:

$$V_{gnd} = \min{(V_{Agnd}, V_{Bgnd})}$$

10.3 Functional Validation

Given the high dependency of many products on internal software, this is probably impossible!