
loticAgent Documentation

lotic

Jun 12, 2020

Contents:

1	Introduction	1
1.1	Iotic Agent In Brief	1
1.2	Getting Started - Installation	1
1.3	Getting Started - Environment - Note on PYTHONPATH	1
1.4	Coding - Minimum script	2
1.5	Coding - Initialisation (ini) file	2
1.6	Coding - Running your script	2
1.7	Coding - Logging	2
1.8	Coding - Wiring up the <i>catchall</i> callback	3
1.9	Coding - Make a thing	3
1.10	Coding - Create a feed on your thing	3
1.11	Coding - Share data from your feed	4
1.12	Coding - Following someone else's feed	4
1.13	Coding - Simulating someone else's feed	4
1.14	Coding - Following one of your own feeds	5
1.15	Coding - Describe your things and feeds in metadata	5
1.16	Coding - Describe the contents of your feed in metadata	5
1.17	Coding - Create a control on your Thing	6
1.18	Coding - Attaching to a remote control	6
1.19	Coding - Which Exceptions	7
2	IoticAgent.Core package	9
2.1	IoticAgent.Core.AmqpLink module	9
2.2	IoticAgent.Core.Const module	10
2.3	IoticAgent.Core.Validation module	10
3	IoticAgent.Datatypes module	13
4	IoticAgent.IOT package	15
4.1	Submodules	15
5	IoticAgent.ThingRunner module	51
6	IoticAgent.Units module	53
7	Indices and tables	57

Python Module Index	59
Index	61

1.1 Iotic Agent In Brief

The *IoticAgent* module is the route from your code to Iotic Space. There's a simple API built around concepts similar to those used in social networking for humans. For example, you can *follow* someone else's feed and *share* your own.

This brief guide shows a quick way to get started coding. It does not attempt to be an exhaustive introduction to all the code. Follow the links on the left to see more detail.

1.2 Getting Started - Installation

The **simplest installation guide** is the [README on GitHub](#).

Head there now to install, then come back here.

1.3 Getting Started - Environment - Note on PYTHONPATH

If you install from Git, or use an advanced pip install technique, you will need to be careful that PYTHONPATH is set to where you've installed the IoticAgent and any dependencies.

- If you used pip to install from PyPI and *didn't* specify a directory in which to install, you don't need to do anything.
- If you cloned github or used pypi with the *-t* option, then you'll need to tell python where to look for the modules you've installed. See the Install from Git instructions [on GitHub](#)

1.4 Coding - Minimum script

Create a new Python script in the src directory with the following lines. This is the minimal Iotic Agent. Let's call it *my_script.py* for now.

```
from IoticAgent import IOT
with IOT.Client() as client:
    pass
```

You can, if you want, specify the config file in the call to *Client()*

```
from IoticAgent import IOT
with IOT.Client(config="my_script.ini") as client:
    pass
```

More details: *IoticAgent.IOT.Client module*

1.5 Coding - Initialisation (ini) file

If you run this script on its own it will cause an exception on start with a helpful message:

```
Exception: Minimum configuration for IoticAgent is epId, passwd and token
```

Create or download from the Iotic Space website file *new_agent/src/my_script.ini* with contents

```
[agent]
epid = x
passwd = y
token = z
```

Where x, y and z are the epid, passwd and token from your owner.

1.6 Coding - Running your script

```
$ python3 my_script.py
```

The minimum script will connect and then wait for incoming messages. CTRL+C will stop it, as it's not very useful at the moment.

1.7 Coding - Logging

The IoticAgent uses the Python logging module to raise messages to the user. Most of these you can ignore and, by default, no logging is shown. To setup logging in your script add the following lines **at the top** of your minimal agent.

```
import logging
logging.basicConfig(format='%(asctime)s, %(msecs)03d %(levelname)s [%(name)s] {
    ↪ %(threadName)s} %(message)s',
                    level=logging.WARNING)
```

This will show Warning, Error and Critical messages only. The next most verbose level is *logging.INFO* which is used to show functions you've called and any extra information a human user might like for normal development. There is one more level of logging (*logging.DEBUG*) which shows all the information about everything which will be too verbose for most people.

1.8 Coding - Wiring up the *catchall* callback

When feeddata arrives at your minimal Agent it will show a message at logging.WARNING level like:

```
Received Feed Data for Point GUID <long-string-of-numbers> but no callback registered.
↪ "
```

The IoticAgent.IOT.Client provides a *catchall* feature for incoming feed data and control request messages that you haven't got round to wiring up yet.

```
def catchall_feeddata(data):
    print("catchall_feeddata:", data)

with IOT.Client(config="my_script.ini") as client:
    client.register_catchall_feeddata(catchall_feeddata)
```

Anytime any feed data arrives at this Agent the *callback_feeddata* (or whatever you want to call it) function will be called.

Note: *client.register_callback_controlreq(fn)* works the same way for control requests.

More details: *IoticAgent.IOT.Client module*

1.9 Coding - Make a thing

You can create a thing programmatically by calling *create_thing()* on your client instance. This method returns an iotic thing object which you should keep for future use.

```
thing_solar_panels = client.create_thing("SolarPanels")
```

This will create you a thing with a "local id" of *SolarPanels*. Think of the *local id* as a nickname for your thing. It is for your use only and only on the container where you created it. No-one else on the system will know this name, be able to search for it or use it.

You can give your thing more descriptive information - see *Describe your things and feeds in metadata* below

More details: *IoticAgent.IOT.Client module*

1.10 Coding - Create a feed on your thing

Now's the time to get your thing to advertise its wares. Call *create_feed()* on your thing instance and give it a *local id*. the same rules apply about this *local id* as before. Keep the returned iotic feed object for use later.

```
feed_current_values = thing_solar_panels.create_feed("Current Values")
```

More details: *IoticAgent.IOT.Thing module*

1.11 Coding - Share data from your feed

Ok, now to the good bit. Your code can now scurry off to find some values to share. Call *share()* on your feed instance. In our example the solar panels' current values feed shares a python dictionary of timestamp, power, current, voltage, and temperature

```
current_values = {}
current_values["timestamp"] = datetime.datetime.now().isoformat()
current_values["power"] = panels.power
current_values["current"] = panels.current
current_values["voltage"] = panels.voltage
current_values["temperature"] = panels.temperature

feed_current_values.share(current_values)
```

You can share pretty much what you like: strings, bytes, dictionaries, json and the Agent will try to bundle it up for you. You can be assured that whatever you share will come out “the other end” in the same format: share a dict; they get a dict, etc.

More details: *IoticAgent.IOT.Point module*

1.12 Coding - Following someone else's feed

To follow someone else's feed, first you need to know the feed's *Globally Unique ID* or *GUID*. You can get this by searching the space on the web UI or they could email it to you. Then you call *follow()* on your thing object for their remote feed. You'll also have to specify the callback function you want the agent to call for you when they next publish that feed.

```
# Args is a dict containing data and other details. See Thing.follow() for more_
↳ information.
def feeddata_cb(args):
    print("feeddata_cb", args['data'])

r_current_values_GUID = "long-string-of-numbers" # other feed's GUID
r_feed_current_values = thing_solar_panels.follow(r_current_values_GUID,
                                                  callback=feeddata_cb)
```

More details: *IoticAgent.IOT.Thing module*

1.13 Coding - Simulating someone else's feed

You might be wondering why you want to remember the RemoteFeed object (*r_feed_current_values* in the code). This can be useful if they don't publish often and you want to simulate their publishes in your code for testing.

Call *simulate()* on the remote feed object like so:

```
try:
    r_feed_current_values.simulate()
except Exception:
    pass # didn't have anything to simulate
```


Your callback function will be called with the last value that was received. The *try/except* block is necessary as *simulate()* will raise an exception if it's got nothing to simulate - normally because you've never received anything and it can't just make something up.

More details: *IoticAgent.IOT.RemotePoint module*

1.14 Coding - Following one of your own feeds

For anything to do with your own things, feeds and controls you can refer to them by their *nickname*. So to follow a feed on one of your own things, call *follow()* with the a tuple containing the nicknames of your thing/feed and your callback.

```
thing_solar_panels_receiver.follow(("SolarPanels", "Current Values"),
                                   callback=feeddata_cb)
```

More details: *IoticAgent.IOT package*

1.15 Coding - Describe your things and feeds in metadata

The Iotic infrastructure makes great use of semantic metadata to provide meaning for your things, Points and your Points' values. You can get a metadata helper object from your Thing (or Feed) and describe the Thing in more detail. The label and description fields are free-text searchable so put information in them that will help others to find them. The geo-location of your Thing is used if others perform a location-based search.

Note: Your metadata is only searchable by others if you call *IoticAgent.IOT.Thing.Thing.set_public*. This allows you some granularity on what you choose to make available to others.

```
#!/python
with thing_solar_panels.get_meta() as meta_thing_solar_panels:
    meta_thing_solar_panels.set_label("Mark's Solar Panels")
    meta_thing_solar_panels.set_description("Solar Array 3.3kW")
    meta_thing_solar_panels.set_location(52.1965071, 0.6067687)

# Optionally set the Thing to be public so that others can search for it.
thing_solar_panels.set_public()
```

More details:

- *IoticAgent.IOT.Thing module*
- *IoticAgent.IOT.ThingMeta module*

1.16 Coding - Describe the contents of your feed in metadata

One of the most useful things you can do in metadata is to let the receiving parties of your feed data know what the contents of your feed are and their data type and units. There are a couple of extra, helpful modules you can import from the IoticAgent **at the top of your code**:

```
from IoticAgent import IOT, Units, Datatypes
```

Then you can specify the contents of your feed in more explicit detail. Call *create_value()* on your feed object.

```
feed_current_values.create_value("timestamp", Datatypes.DATETIME,
                                "en", "time of reading")
feed_current_values.create_value("power", Datatypes.DECIMAL,
                                "en", "output power in watts",
                                Units.WATT)
feed_current_values.create_value("current", Datatypes.DECIMAL,
                                "en", "dc current in amps",
                                Units.AMPERE)
feed_current_values.create_value("voltage", Datatypes.DECIMAL,
                                "en", "dc potential in volts",
                                Units.VOLT)
feed_current_values.create_value("temperature", Datatypes.DECIMAL,
                                "en", "array temperature in celsius",
                                Units.CELSIUS)
```

More details: *IoticAgent.IOT.Point module*

1.17 Coding - Create a control on your Thing

You can also make your Thing respond to activation requests by creating a *control* and its associated callback. Call *create_control* on your Thing object. A control like the opposite of a feed. It's a way that other Things can send you data or ask your Thing to perform some action(s).

```
def reset_callback(args):
    print("resetting counters")
    panels.zero_counters()

thing_solar_panels.create_control("Reset Counters", reset_callback)
```

When somebody else activates your control with an *ask()* or *tell()*, then your callback will be called.

args is a dict containing control data and other details. See *IoticAgent.IOT.Thing module* for more information.

1.18 Coding - Attaching to a remote control

Attaching to a remote control is a bit like *following*-ing a feed, except the functions is *attach()* on the remote Thing. Once you've attached, you can call *ask()* or *tell()* on your RemoteControl object.

```
r_counters_GUID = "long-string-of-numbers" # GUID of remote control
r_control_counters = thing_solar_panels.attach(r_counters_GUID)
...
r_control_counters.ask(<payload>) #activate the remote control
```

The two similar functions *ask()* and *tell()* differ in that *ask()* is doesn't require a confirmation that the action has been done, but *tell()* has a callback to be informed when the far end has done the action and a time-out so you can be informed if they haven't done it in the time you require.

As you might expect, the *attach* function works for controls on Things you own. Use the tuple of (*thing_nickname, control_nickname*) to identify the control instead of the GUID

1.19 Coding - Which Exceptions

Parameters are validated on request and raise built-in `ValueError`.

Network failures raise *IoticAgent.Core.Exceptions.LinkException* which can be imported from *IoticAgent.IOT.Exceptions* for convenience.

If a request response from Iotic Space contains (FAILED)

IoticAgent.IOT.Exceptions.IOTUnknown (Unknown resource EG Thing not found)

IoticAgent.IOT.Exceptions.IOTMalformed (Not allowed resource, malformed request)

IoticAgent.IOT.Exceptions.IOTInternalError (Container internal error)

IoticAgent.IOT.Exceptions.IOTAccessDenied (ACL restriction)

IoticAgent.Core package

IoticAgent.Core module provides a Thread Safe AMQP Client connection to the Iotic Labs QAPI

Calling a request function returns a Core.RequestEvent instance on which the caller can .wait Or check using the RequestEvent.is_set function

When the event is complete event.requestId, success and payload will be populated. RequestEvent._messages will contain the raw messages from the queue.

2.1 IoticAgent.Core.AmqpLink module

```
class IoticAgent.Core.AmqpLink.AmqpLink(host, vhost, prefix, epid, passwd,
                                         msg_callback, ka_callback, send_ready_callback,
                                         sslca=None, prefetch=128, ackpc=0.5,
                                         heartbeat=30, socket_timeout=10,
                                         startup_ignore_exc=False, conn_retry_delay=5,
                                         conn_error_log_threshold=180)
```

Bases: object

Helper class to deal with AMQP connection.

is_alive()

Helper function to show if send & recv Threads are running

last_send_exc_time

Timestamp (or None) at which send thread last failed

send(body, content_type='application/ubjson', timeout=5)

timeout indicates amount of time to wait for sending thread to be ready. set to larger than zero to wait (in seconds, fractional) or None to block.

start()

start connection threads, blocks until started

stop()

disconnect, blocks until stopped

2.2 IoticAgent.Core.Const module

QAPI Constants

class IoticAgent.Core.Const.DescribeScope

Bases: enum.Enum

Determines how a resource is described

AUTO - Uses local meta if available, otherwise public

PUBLIC - Public metadata (public)

LOCAL - Container-local meta (only available if container running in ‘private’ mode)

LOCAL_OWN - Container-local meta for own resource only

AUTO = 'auto'

LOCAL = 'local'

LOCAL_OWN = 'localOwn'

PUBLIC = 'public'

class IoticAgent.Core.Const.SearchScope

Bases: enum.Enum

Determines how search is performed

PUBLIC - Public metadata search

LOCAL - Container-local search (only available if container running in ‘private’ mode)

LOCAL_OWN - Container-local search for own things only

LOCAL = 'local'

LOCAL_OWN = 'localOwn'

PUBLIC = 'public'

class IoticAgent.Core.Const.SearchType

Bases: enum.Enum

Determines what results a search returns

FULL - Provides most information

REDUCED - Limit output including only points and their type

LOCATED - Minimal location + label format (e.g. for display on a map)

FULL = 'full'

LOCATED = 'located'

MINIMAL = 'minimal'

REDUCED = 'reduced'

2.3 IoticAgent.Core.Validation module

class IoticAgent.Core.Validation.Validation

Bases: object

static bool_check_convert (*bname, barg*)

Currently does no checking - passthrough for now in case people change their minds

static callable_check (*func, arg_count=1, arg_value=None, allow_none=False*)

Check whether func is callable, with the given number of positional arguments.

Returns True if check succeeded, False otherwise.

static check_convert_string (*obj, name=None, no_leading_trailing_whitespace=True, no_whitespace=False, no_newline=True, as_tag=False, min_len=1, max_len=0*)

Ensures the provided object can be interpreted as a unicode string, optionally with additional restrictions imposed. By default this means a non-zero length string which does not begin or end in whitespace.

classmethod comment_check_convert (*comment, allow_none=False*)

classmethod datetime_check_convert (*time, allow_none=False, require_utc=True, to_iso8601=True*)

classmethod describe_scope_check_convert (*type_*)

classmethod description_check_convert (*comment, allow_none=False*)

static foc_check (*foc*)

static guid_check_convert (*guid, allow_none=False*)

Take a GUID in the form of hex string "32" or "8-4-4-4-12".

Returns badly formed hexadecimal UUID string

Return type Hex string "32" or raises ValueError

classmethod label_check_convert (*label, allow_none=False*)

classmethod lang_check_convert (*lang, allow_none=False, default=None*)

classmethod lid_check_convert (*lid*)

static limit_offset_check (*limit, offset*)

static location_check (*lat, lon*)

For use by Core client wrappers

static metafmt_check_convert (*fmt*)

classmethod mime_check_convert (*mime, allow_none=False*)

static object_type_check_convert (*otype*)

classmethod pid_check_convert (*lid*)

classmethod proplist_check_convert (*props, for_delete=False*)

for_delete allows a completely empty list as well as no value for individual properties. This is for deletion of all properties or removal of all properties with a given predicate respectively.

classmethod search_check_convert (*text, lang, location, unit, default_lang*)

classmethod search_location_check (*location*)

Core.Client.request_search location parameter should be a dictionary that contains lat, lon and radius floats

classmethod search_scope_check_convert (*type_*)

classmethod search_type_check_convert (*type_*)

classmethod tags_check_convert (*tags*)

Accept one tag as string or multiple tags in list of strings.

Returns List (with tags in unicode form) or raises ValueError

```
static uriref_check_convert (ref, name='uriref')  
    Performs minimal/sanity validation for RDF URI References.  
  
classmethod value_type_check_convert (vtype)  
  
classmethod value_unit_check_convert (unit)
```

IoticAgent.Datatypes module

Constants to hide XSD Datatypes used by Point Values and Properties These help to describe the data in a feed so the receiving Thing can know what kind of data to expect

See also <http://www.w3.org/TR/xmlschema-2/#built-in-datatypes>

`IoticAgent.Datatypes.BASE64 = 'base64Binary'`

Represents a sequence of binary octets (bytes) encoded according to RFC 2045, the standard defining the MIME types (look under “6.8 Base64 Content-Transfer-Encoding”).

`IoticAgent.Datatypes.BOOLEAN = 'boolean'`

A Boolean true or false value. Representations of true are “true” and “1”; false is denoted as “false” or “0”.

`IoticAgent.Datatypes.BYTE = 'byte'`

A signed 8-bit integer in the range [-128 -> +127]. Derived from the short datatype.

`IoticAgent.Datatypes.DATE = 'date'`

Represents a specific date. The syntax is the same as that for the date part of dateTime, with an optional time zone indicator. Example: “1889-09-24”.

`IoticAgent.Datatypes.DATETIME = 'dateTime'`

ss followed by an optional time-zone suffix.

YYYY is the year, *MM* is the month number, *DD* is the day number, *hh* the hour in 24-hour format, *mm* the minute, and *ss* the second (a decimal and fraction are allowed for the seconds part).

The optional zone suffix is either “Z” for Universal Coordinated Time (UTC), or a time offset of the form “[+|-]*hh:mm*”, giving the difference between UTC and local time in hours and minutes.

Example: “2004-10-31T21:40:35.5-07:00” is a time on Halloween 2004 in Mountain Standard time. The equivalent UTC would be “2004-11-01T04:40:35.5Z”.

Type Represents a specific instant of time. It has the form *YYYY-MM-DDThh*

Type *mm*

`IoticAgent.Datatypes.DECIMAL = 'decimal'`

Any base-10 fixed-point number. There must be at least one digit to the left of the decimal point, and a leading “+” or “-” sign is allowed. Examples: “42”, “-3.14159”, “+0.004”.

`IoticAgent.Datatypes.DOUBLE = 'double'`

A 64-bit floating-point decimal number as specified in the IEEE 754-1985 standard. The external form is the same as the float datatype.

`IoticAgent.Datatypes.FLOAT = 'float'`

A 32-bit floating-point decimal number as specified in the IEEE 754-1985 standard. Allowable values are the same as in the decimal type, optionally followed by an exponent, or one of the special values “INF” (positive infinity), “-INF” (negative infinity), or “NaN” (not a number).

The exponent starts with either “e” or “E”, optionally followed by a sign, and one or more digits.

Example: “6.0235e-23”.

`IoticAgent.Datatypes.INT = 'int'`

Represents a 32-bit signed integer in the range [-2,147,483,648, 2,147,483,647]. Derived from the long datatype.

`IoticAgent.Datatypes.INTEGER = 'integer'`

Represents a signed integer. Values may begin with an optional “+” or “-” sign. Derived from the decimal datatype.

`IoticAgent.Datatypes.IRI = 'IRI'`

Only for use with property API calls. Used to handle properties which require an IRI (URIRef) value.

`IoticAgent.Datatypes.LONG = 'long'`

A signed, extended-precision integer; at least 18 digits are guaranteed. Derived from the integer datatype.

`IoticAgent.Datatypes.STRING = 'string'`

Any sequence of zero or more characters.

`IoticAgent.Datatypes.TIME = 'time'`

A moment of time that repeats every day. The syntax is the same as that for `dateTime`, omitting everything up to and including the separator “T”. Examples: “00:00:00” is midnight, and “13:04:00” is an hour and four minutes after noon.

`IoticAgent.Datatypes.UNSIGNED_BYTE = 'unsignedByte'`

An unsigned 8-bit integer in the range [0, 255]. Derived from the `unsignedShort` datatype.

`IoticAgent.Datatypes.URI = 'anyURI'`

The data must conform to the syntax of a Uniform Resource Identifier (URI), as defined in RFC 2396 as amended by RFC 2732. Example: “<http://www.nmt.edu/tcc/>” is the URI for the New Mexico Tech Computer Center’s index page.

IoticAgent.IOT package

IoticAgent.IOT module provides a simple wrapper for IoticAgent.Core with helper objects for Thing, Point, etc.

4.1 Submodules

4.1.1 IoticAgent.IOT.Client module

The Client object used to initiate connection to Iotic Space

```
class IoticAgent.IOT.Client.Client (config=None)  
    Bases: object
```

```
    agent_id  
        Agent id (aka epId) in use for this client instance
```

```
    confirm_tell (data, success)  
        Confirm that you've done as you were told. Call this from your control callback to confirm action. Used  
        when you are advertising a control and you want to tell the remote requestor that you have done what they  
        asked you to.
```

Example

This is a minimal example to show the idea.

Note: No Exception handling and ugly use of globals

```
client = None  
  
def controlreq_cb(args):  
    global client    # the client object you connected with  
  
    # perform your action with the data they sent
```

(continues on next page)

(continued from previous page)

```

success = do_control_action(args['data'])

if args['confirm']: # you've been asked to confirm
    client.confirm_tell(args, success)
# else, if you do not confirm_tell() this causes a timeout at the_
↪requestor's end.

client = IOT.Client(config='test.ini')
thing = client.create_thing('test321')
control = thing.create_control('control', controlreq_cb)

```

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communication problem between you and the infrastructure

Parameters

- **data** (*dictionary*) – The “args” dictionary that your callback was called with
- **success** (*boolean*) – Whether or not the action you have been asked to do has been successful

More details on the contents of the *data* dictionary for controls see *IoticAgent.IOT.Thing module* `create_control`.

create_thing (lid)

Create a new Thing with a local id (lid).

Returns A *IoticAgent.IOT.Thing module* Thing object if successful or if the Thing already exists

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters **lid** (*string*) – local identifier of your Thing. The local id is your name or nickname for the thing. It’s “local” in that it’s only available to you on this container, not searchable and not visible to others.

create_thing_async (lid)**default_lang**

Language in use when not explicitly specified (in meta related requests). Will be set to container default if was not set in configuration. Before client has started this might be None.

delete_thing (lid)

Delete a Thing

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

lid (string) local identifier of the Thing you want to delete

delete_thing_async (lid)**describe (guid_or_resource, lang=None, scope=<DescribeScope.AUTO: 'auto'>)**

Describe returns the public (or local) description of a Thing or Point

Returns The description dict (see below for Thing example) if available, otherwise *None*

```
{
  "type": "Entity",
  "meta": {
    "long": 0.716356,
    "lat": 52.244384,
    "label": "Weather Station #1",
    "parent": "3bbf307b43b1460289fe707619dece3d",
    "points": [
      {
        "type": "Control",
        "label": "Control 101",
        "guid": "fb1a4a4dbb2642ab9f836892da93c101",
        "storesRecent": false
      },
      {
        "type": "Feed",
        "label": "My weather feed",
        "guid": "fb1a4a4dbb2642ab9f836892da93f101",
        "storesRecent": true
      }
    ],
    "comment": "A lovely weather station...",
    "tags": [
      "blue",
      "garden"
    ]
  }
}
```

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **guid_or_resource** (*string or object*) – If a *string*, it should contain the globally unique id of the resource you want to describe in 8-4-4-12 (or undashed) format. If an *object*, it should be an instance of `Thing`, `Point`, `RemoteFeed` or `RemoteControl`. The system will return you the description of that object.
- **lang** (*string, optional*) – The two-character ISO 639-1 language code for which labels and comments will be returned. This does not affect Values (i.e. when describing a `Point`, apart from value comments) and tags as these are language neutral).
- **scope** (*optional*) – Whether to perform `PUBLIC`, `LOCAL` (container level) or `LOCAL_OWN` (container level restricted to own things) metadata lookup. Check the [IoticAgent.IOT.Client module](#) `Client.local_meta` flag to determine whether local metadata functionality is available. (Note that `AUTO`, `PUBLIC` and `LOCAL_OWN` scopes are always available.). `AUTO` mode first attempts to look up private metadata, then public.

get_thing (*lid*)

Get the details of a newly created Thing. This only applies to asynchronous creation of Things and the new Thing instance can only be retrieved once.

Returns A *IoticAgent.IOT.Thing module* Thing object, which corresponds to the Thing with the given local id (nickname)

Raises `KeyError` – The Thing has not been newly created (or has already been retrieved by a previous call)

Parameters `lid` (*string*) – local identifier of your Thing.

`is_connected()`

Returns Client's alive state

`list` (*all_my_agents=False, limit=500, offset=0*)

List *all* the things created by this client on this or all your agents

Returns QAPI list function payload

Raises `LinkException` – Communications problem between you and the infrastructure

Parameters

- **all_my_agents** (*boolean, optional*) – If *False* limit search to just this agent, if *True* return list of things belonging to all agents you own.
- **limit** (*integer, optional*) – Return this many Point details
- **offset** (*integer, optional*) – Return Point details starting at this offset

`local_meta`

Whether container-local metadata functionality (e.g. search) is available in this container. Before calling `start()` this will always be *False*.

`register_callback_created` (*func, serialised=True*)

Register a callback for resource creation. This will be called when any *new* resource is created within your agent. If *serialised* is not set, the callbacks might arrive in a different order to they were requested.

The payload passed to your callback is an `OrderedDict` with the following keys

<code>r</code>	:	<code>R_ENTITY, R_FEED, etc</code>	# the type of resource created
<code>lid</code>	:	<code><name></code>	# the local name of the resource
<code>id</code>	:	<code><GUID></code>	# the global Id of the resource
<code>epId</code>	:	<code><GUID></code>	# the global Id of your agent

Note: Resource types are defined in *IoticAgent.Core.Const module*

Example

```
def created_callback(args):
    print(args)
...
client.register_callback_created(created_callback)
```

This would print out something like the following on creation of an `R_ENTITY`

```
OrderedDict([(u'lid', u'new_thing1'), (u'r', 1),
             (u'epId', u'ffd47b75ea786f55c76e337cdc47665a'),
             (u'id', u'3f11df0a09588a6a1a9732e3837765f8')]))
```

`register_callback_deleted` (*func, serialised=True*)

Register a callback for resource deletion. This will be called when any resource is deleted within your agent. If *serialised* is not set, the callbacks might arrive in a different order to they were requested.

The payload passed to your callback is an OrderedDict with the following keys

```
r      : R_ENTITY, R_FEED, etc # the type of resource deleted
lid    : <name>                 # the local name of the resource
id     : <GUID>                 # the global Id of the resource
```

Note: Resource types are defined in *IoticAgent.Core.Const module*

Example

```
def deleted_callback(args):
    print(args)
...
client.register_callback_deleted(deleted_callback)
```

This would print out something like the following on deletion of an R_ENTITY

```
OrderedDict([(u'lid', u'old_thing1'),
             (u'r', 1),
             (u'id', u'315637813d801ec6f057c67728bf00c2')])
```

register_callback_duplicate (*func*, *serialised=True*)

Register a callback for resource creation but where the resource already exists in Iotic Space. In this case the existing reference is passed to you. If *serialised* is not set, the callbacks might arrive in a different order to they were requested.

The payload passed to your callback is an OrderedDict with the following keys:

```
r      : R_ENTITY, R_FEED, etc # the type of existing resource
lid    : <name>                 # the local name of the
                                   # existing resource
id     : <GUID>                 # the global Id of the
                                   # existing resource
epId   : <GUID>                 # the global Id of your agent
```

Note: Resource types are defined in *IoticAgent.Core.Const module*

Example

```
def duplicated_callback(args):
    print(args)
...
client.register_callback_created(duplicated_callback)
```

This would print out something like the following on re-creation of an R_ENTITY

```
OrderedDict([(u'lid', u'new_thing1'), (u'r', 1),
             (u'epId', u'ffd47b75ea786f55c76e337cdc47665a'),
             (u'id', u'3f11df0a09588a6a1a9732e3837765f8')])
```

register_callback_reassigned (*func*, *serialised=True*)

Register a callback for resource reassignment. This will be called when any resource is reassigned to or from your agent. If *serialised* is not set, the callbacks might arrive in a different order to they were requested.

The payload passed to your callback is an OrderedDict with the following keys:

```
r      : R_ENTITY, R_FEED, etc # the type of resource reassigned
lid    : <name>                # the local name of the resource
epId   : <GUID>                # the global Id of the agent the
                                     # resource has been reassigned *to*
id     : <GUID>                # the global Id of the resource
```

Note: Resource types are defined in *IoticAgent.Core.Const module*

Note: You can check whether this is an assign “in” or “out” by comparing the epId with your current agent id, using the *IOT.Client.agent_id* property. If it’s the same it’s a reassign to you.

Example

```
def reassigned_callback(args):
    print(args)
...
client.register_callback_reassigned(reassigned_callback)
```

This would print out something like the following on assignment of an R_ENTITY to:

```
OrderedDict([(u'lid', u'moved_thing'),
              (u'r', 1),
              (u'epId', u'5a8d603ee757133d66d99875d0584c72'),
              (u'id', u'4448993b44738411de5fe2a6cf32d957')])
```

register_callback_renamed (*func, serialised=True*)

Register a callback for resource rename. This will be called when any resource is renamed within your agent.

If *serialised* is not set, the callbacks might arrive in a different order to they were requested.

The payload passed to your callback is an OrderedDict with the following keys

```
r      : R_ENTITY, R_FEED, etc # the type of resource deleted
lid    : <name>                # the new local name of the resource
oldLid  : <name>                # the old local name of the resource
id     : <GUID>                # the global Id of the resource
```

Note: Resource types are defined in *IoticAgent.Core.Const module*

Example

```
def renamed_callback(args):
    print(args)
...
client.register_callback_renamed(renamed_callback)
```

This would print out something like the following on renaming of an R_ENTITY

```
OrderedDict([(u'lid', u'new_name'),
              (u'r', 1),
```

(continues on next page)

(continued from previous page)

```
(u'oldLid', u'old_name'),
(u'id', u'4448993b44738411de5fe2a6cf32d957'))]
```

register_callback_subscribed (*callback*)

Register a callback for new subscription. This gets called whenever one of *your* things subscribes to something else.

Note: It is not called when whenever something else subscribes to your thing.

The payload passed to your callback is either a *IoticAgent.IOT.RemotePoint module* RemoteControl or RemoteFeed instance.

register_callback_subscription (*callback*)

Register a callback for subscription count change notification. This gets called whenever something *else* subscribes to your thing.

Note: It is not called when you subscribe to something else.

The payload passed to your callback is an OrderedDict with the following keys

```
r          : R_FEED or R_CONTROL # the type of the point
lid        : <name>               # the local name of your *Point*
entityLid  : <name>               # the local name of your *Thing*
subCount   : <count>              # the total number of remote Things
                                     # that subscribe to your point
```

Example

```
def subscription_callback(args):
    print(args)
    ...
client.register_callback_subscription(subscription_callback)
```

This would print out something like the following

```
OrderedDict([('r', 2), ('lid', 'My_First_Point'),
             ('entityLid', 'My_First_Thing'), ('subCount', 13)])
```

register_catchall_controlreq (*callback*, *callback_parsed=None*)

Registers a callback that is called for all control requests received by your Thing

Example

```
def controlreq_callback(data):
    print(data)
    ...
client.register_catchall_controlreq(controlreq_callback)
```

Parameters

- **callback** – the function name that you want to be called on receipt of a new control request

- **callback_parsed** (*function reference, optional*) – callback function to invoke on receipt of a control ask/tell. This is equivalent to *callback* except the dict includes the *parsed* key which holds the set of values in a *IoticAgent.IOT.Point module* *PointDataObject* instance. If both *callback_parsed* and *callback* have been specified, the former takes precedence and *callback* is only called if the point data could not be parsed according to its current value description.

More details on the contents of the *data* dictionary for controls see *IoticAgent.IOT.Thing module* *create_control*

register_catchall_feedata (*callback, callback_parsed=None*)
Registers a callback that is called for all feeddata your Thing receives

Example

```
def feeddata_callback(data):  
    print(data)  
...  
client.register_catchall_feedata(feeddata_callback)
```

Parameters

- **callback** – the function name that you want to be called on receipt of new feed data
- **callback_parsed** (*function reference, optional*) – callback function to invoke on receipt of feed data. This is equivalent to *callback* except the dict includes the *parsed* key which holds the set of values in a *IoticAgent.IOT.Point module* *PointDataObject* instance. If both *callback_parsed* and *callback* have been specified, the former takes precedence and *callback* is only called if the point data could not be parsed according to its current value description.

Note: *callback_parsed* can only be used if *auto_encode_decode* is enabled for the client instance.

More details on the contents of the *data* dictionary for feeds see *IoticAgent.IOT.Thing module* *Thing.follow*.

save_config ()
Save the config, update the seqnum & default language

search (*text=None, lang=None, location=None, unit=None, limit=50, offset=0, reduced=False, scope=<SearchScope.PUBLIC: 'public'>*)
Search the Iotic Space for public Things with metadata matching the search parameters: text, lang(uage), location, unit, limit, offset. Note that only things which have at least one point defined can be found.

Returns Dict of results as below (first with *reduced=False*, second with *reduced=True*)

```
# reduced=False returns dict similar to below  
{  
    "2b2d8b068e404861b19f9e060877e002": {  
        "long": -1.74803,  
        "lat": 52.4539,  
        "label": "Weather Station #2",  
        "owner": "3bbf307b43b1460289fe707619dece3d",  
        "points": {  
            "a300cc90147f4e2990195639de0af201": {  
                "label": "Feed 201",  
                "type": "Feed",
```

(continues on next page)

(continued from previous page)

```

        "storesRecent": true
    },
    "a300cc90147f4e2990195639de0af202": {
        "label": "Feed 202",
        "type": "Feed",
        "storesRecent": false
    }
},
"76a3b24b02d34f20b675257624b0e001": {
    "long": 0.716356,
    "lat": 52.244384,
    "label": "Weather Station #1",
    "owner": "3bbf307b43b1460289fe707619dece3d",
    "points": {
        "fb1a4a4dbb2642ab9f836892da93f101": {
            "label": "My weather feed",
            "type": "Feed",
            "storesRecent": false
        },
        "fb1a4a4dbb2642ab9f836892da93c102": {
            "label": None,
            "type": "Control",
            "storesRecent": false
        }
    }
}
}

```

```

# reduced=True returns dict similar to below
{
    "2b2d8b068e404861b19f9e060877e002": {
        "a300cc90147f4e2990195639de0af201": "Feed",
        "a300cc90147f4e2990195639de0af202": "Feed"
    },
    "76a3b24b02d34f20b675257624b0e001": {
        "fb1a4a4dbb2642ab9f836892da93f101": "Feed",
        "fb1a4a4dbb2642ab9f836892da93f102": "Control"
    }
}

```

OR**Raises**

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **text** (*string, optional*) – The text to search for. Label and description will be searched for both Thing and Point and each word will be used as a tag search too. Text search is case-insensitive. Tag search is language neutral.
- **lang** (*string, optional*) – The two-character ISO 639-1 language code to search in, e.g. “en” “fr”. Language is used to limit search to only labels and descriptions in that language. You will only get labels *in that language* back from search and then only if there are any in that language.

- **location** (*dictionary, optional*) – Latitude, longitude and radius to search within. All values are float, Radius is in kilometers (km). E.g. {"lat":1.2345, "long":54.321, "radius":6.789}. If *text* has not been specified, radius can at most be 25km.
- **unit** (*string, optional*) – Valid URL of a unit in an ontology. Or use a constant from the *IoticAgent.Units module* class
- **limit** (*integer, optional*) – Return this many search results
- **offset** (*integer, optional*) – Return results starting at this offset - good for paging.
- **reduced** (*boolean, optional*) – If *true*, return the reduced results just containing points and their type.
- **scope** (*optional*) – Whether to perform PUBLIC, LOCAL (container level) or LOCAL_OWN (container level restricted to own things) search. Check the *IoticAgent.IOT.Client module* Client.local_meta flag to determine whether local metadata functionality is available. (Note that PUBLIC and LOCAL_OWN scopes are always available.)

search_located (*text=None, lang=None, location=None, unit=None, limit=100, offset=0, scope=<SearchScope.PUBLIC: 'public'>*)

See *IoticAgent.IOT.Client module* Client.search for general documentation. Provides a thing-only result set comprised only of things which have a location set, e.g.:

```
{
  # Keyed by thing id
  '2b2d8b068e404861b19f9e060877e002':
    # location (g, lat & long), label (l, optional)
    {'g': (52.4539, -1.74803), 'l': 'Weather Station #2'},
  '76a3b24b02d34f20b675257624b0e001':
    {'g': (52.244384, 0.716356), 'l': None},
  '76a3b24b02d34f20b675257624b0e004':
    {'g': (52.245384, 0.717356), 'l': 'Gasometer'},
  '76a3b24b02d34f20b675257624b0e005':
    {'g': (52.245384, 0.717356), 'l': 'Zepellin'}
```

search_minimal (*text=None, lang=None, location=None, unit=None, limit=100, offset=0, scope=<SearchScope.PUBLIC: 'public'>*)

See *IoticAgent.IOT.Client module* Client.search for general documentation. Returns just a list of the id's of matching things.

```
[
  '2b2d8b068e404861b19f9e060877e002',
  '76a3b24b02d34f20b675257624b0e001',
  '76a3b24b02d34f20b675257624b0e004',
  '76a3b24b02d34f20b675257624b0e005',
]
```

search_property (*props, limit=50, offset=0, type_=<SearchType.FULL: 'full'>, scope=<SearchScope.PUBLIC: 'public'>, lang=None, with_pointless=False*)

Search the Iotic Space for public Things with metadata matching the properties list, given as predicate-object tuples, with a third member denoting an XSD type if the object type is not among [str, int, float, bool]. Language-specific strings are specified with a suffix of two @'s and a two-letter language code, e.g. '@@en'.

Returns Dict of results as below (with type_=SearchType.FULL; see shorthand methods for examples of other search type results)

```
{
  "2b2d8b068e404861b19f9e060877e002": {
    "long": -1.74803,
    "lat": 52.4539,
    "label": "Weather Station #2",
    "owner": "3bbf307b43b1460289fe707619dece3d",
    "points": {
      "a300cc90147f4e2990195639de0af201": {
        "label": "Feed 201",
        "type": "Feed",
        "storesRecent": true
      },
      "a300cc90147f4e2990195639de0af202": {
        "label": "Feed 202",
        "type": "Feed",
        "storesRecent": false
      }
    }
  },
  "76a3b24b02d34f20b675257624b0e001": {
    "long": 0.716356,
    "lat": 52.244384,
    "label": "Weather Station #1",
    "owner": "3bbf307b43b1460289fe707619dece3d",
    "points": {
      "fb1a4a4dbb2642ab9f836892da93f101": {
        "label": "My weather feed",
        "type": "Feed",
        "storesRecent": false
      },
      "fb1a4a4dbb2642ab9f836892da93c102": {
        "label": None,
        "type": "Control",
        "storesRecent": false
      }
    }
  }
}
```

OR

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **props** (*list (tuple)*) – The properties (or single property) to search for
- **limit** (*integer, optional*) – Return this many search results
- **offset** (*integer, optional*) – Return results starting at this offset - good for paging.
- **type** (*optional*) – What kind of search to perform, FULL, REDUCED, LOCATED, or MINIMAL. See above for sample FULL results. See shorthand methods for sample REDUCED, LOCATED, and MINIMAL results.
- **scope** (*optional*) – Whether to perform PUBLIC, LOCAL (container level) or

LOCAL_OWN (container level restricted to own things) search. Check the *IoticAgent.IOT.Client module* Client.local_meta flag to determine whether local metadata functionality is available. (Note that PUBLIC and LOCAL_OWN scopes are always available.)

- **lang** (*string, optional*) – The language in which to return entity and point labels. Defaults to the metadata library's default language.
- **with_pointless** (*bool, optional*) – Whether to return entities that do not advertise any points (default False)

search_property_located (*props, limit=100, offset=0, scope=<SearchScope.PUBLIC: 'public'>, lang=None, with_pointless=False*)

Shorthand for `search_property()` from the *IoticAgent.IOT.Client module* with `type_=SearchType.LOCATED`. Provides a thing-only result set comprised only of things which have a location set, e.g.:

```
{
  # Keyed by thing id
  '2b2d8b068e404861b19f9e060877e002':
    # location (g, lat & long), label (l, optional)
    {'g': (52.4539, -1.74803), 'l': 'Weather Station #2'},
  '76a3b24b02d34f20b675257624b0e001':
    {'g': (52.244384, 0.716356), 'l': None},
  '76a3b24b02d34f20b675257624b0e004':
    {'g': (52.245384, 0.717356), 'l': 'Gasometer'},
  '76a3b24b02d34f20b675257624b0e005':
    {'g': (52.245384, 0.717356), 'l': 'Zepellin'}
```

search_property_minimal (*props, limit=100, offset=0, scope=<SearchScope.PUBLIC: 'public'>, with_pointless=False*)

Shorthand for `search_property()` from the *IoticAgent.IOT.Client module* with `type_=SearchType.MINIMAL`. Returns just a list of the id's of matching things.

```
[
  '2b2d8b068e404861b19f9e060877e002',
  '76a3b24b02d34f20b675257624b0e001',
  '76a3b24b02d34f20b675257624b0e004',
  '76a3b24b02d34f20b675257624b0e005',
]
```

search_property_reduced (*props, limit=100, offset=0, scope=<SearchScope.PUBLIC: 'public'>, with_pointless=False*)

Shorthand for `search_property()` from the *IoticAgent.IOT.Client module* with `type_=SearchType.REDUCED`. Sample results:

```
{
  "2b2d8b068e404861b19f9e060877e002": {
    "a300cc90147f4e2990195639de0af201": "Feed",
    "a300cc90147f4e2990195639de0af202": "Feed"
  },
  "76a3b24b02d34f20b675257624b0e001": {
    "fb1a4a4dbb2642ab9f836892da93f101": "Feed",
    "fb1a4a4dbb2642ab9f836892da93f102": "Control"
  }
}
```

search_reduced (*text=None, lang=None, location=None, unit=None, limit=100, offset=0, scope=<SearchScope.PUBLIC: 'public'>*)

Shorthand for `search()` from the *IoticAgent.IOT.Client module* with `reduced=True`

simulate_feeddata (*feedid, data, mime=None, time=None*)

Simulate the last feeddata received for given feedid. Calls the registered callback for the feed with the last recieved feed data. Allows you to test your code without having to wait for the remote thing to share again.

Parameters

- **feedid** (*string*) – local id of your Feed
- **data** (*optional*) – The data you want to use to simulate the arrival of remote feed data
- **mime** (*string, optional*) – The mime type of your data. See also *IoticAgent.IOT.Point module* `Feed.share`
- **time** (*datetime, optional*) – UTC timestamp for share. See also: *IoticAgent.IOT.Point module* `Feed.share`

start ()

Open a connection to Iotic Space. `start()` is called by `__enter__` which allows the python *with* syntax to be used.

Example 0:

Calling `start()` implicitly using `with`. Minimal example with no exception handling:

```
with IOT.Client(config="my_script.ini") as client:
    client.create_thing("my_thing")
```

Example 1

Calling `start()` implicitly using `with`. This handles the finally for you. *Recommended:*

```
try:
    with IOT.Client(config="my_script.ini") as client:
        try:
            client.create_thing("my_thing")
        except IOTException as exc:
            # handle exception
except Exception as exc: # not possible to connect
    print(exc)
import sys
sys.exit(1)
```

Example 2

Calling `start()` explicitly (no `with`). Be careful, you have to put a finally in your try blocks otherwise your client might remain connected.

```
try:
    client = IOT.Client(config="my_script.ini")
    # wire up callbacks here
    client.start()
except Exception as exc:
    print(exc)
import sys
sys.exit(1)

try:
    client.create_thing("my_thing")
except IOTException as exc:
```

(continues on next page)

(continued from previous page)

```
# handle individual exception
finally:
    client.stop()
```

Returns This Client instance

Raises Exception

stop()

Close the connection to Iotic Space. *stop()* is called by `__exit()` allowing the python *with* syntax to be used. See [IoticAgent.IOT.Client module](#) client start

sync_timeout

Value of `iot.sync_request_timeout` configuration option. Used by all synchronous requests to limit total request wait time.

4.1.2 IoticAgent.IOT.Config module

Wrapper object for the Client Configuration file

class `IoticAgent.IOT.Config.Config` (*fn=None, string=None*)

Bases: `object`

get (*section, val*)

Get a setting or the default

Returns The current value of the setting *val* or the default, or *None* if not found

section (string) the section name in the config E.g. “agent”

val (string) the section name in the config E.g. “host”

save (*filename=None*)

Write config to file.

set (*section, val, data*)

Add a setting to the config

section (string) the section name in the config E.g. “agent”

val (string) the section name in the config E.g. “host”

data the new value for the *val*

setup_logging()

Setup logging module based on known modules in the config file

update (*section, val, data*)

Add a setting to the config, but if same as default or *None* then no action. This saves the .save writing the defaults

section (string) the section name in the config E.g. “agent”

val (string) the section name in the config E.g. “host”

data the new value for the *val*

`IoticAgent.IOT.Config.dict_to_cp` (*dic*)

`IoticAgent.IOT.Config.str_to_logging` (*level*)

4.1.3 IoticAgent.IOT.Exceptions module

exception IoticAgent.IOT.Exceptions.**IOTAccessDenied** (*msg*, *event=None*)

Bases: *IoticAgent.IOT.Exceptions.IOTException*

Request FAILED with ACL Access Denied

exception IoticAgent.IOT.Exceptions.**IOTClientError** (*msg*, *event=None*)

Bases: *IoticAgent.IOT.Exceptions.IOTException*

Unexpected agent-local failure

exception IoticAgent.IOT.Exceptions.**IOTException** (*msg*, *event=None*)

Bases: *Exception*

Base Exception class for IOT

event

RequestEvent instance associated with exception, or None if not available/applicable

Type Returns

exception IoticAgent.IOT.Exceptions.**IOTInternalError** (*msg*, *event=None*)

Bases: *IoticAgent.IOT.Exceptions.IOTException*

Request FAILED with Internal Error

exception IoticAgent.IOT.Exceptions.**IOTMalformed** (*msg*, *event=None*)

Bases: *IoticAgent.IOT.Exceptions.IOTException*

Request FAILED due to invalid parameters (this might indicate agent incompatibility)

exception IoticAgent.IOT.Exceptions.**IOTNotAllowed** (*msg*, *event=None*)

Bases: *IoticAgent.IOT.Exceptions.IOTException*

Request FAILED to unsupporter request (e.g. a feature is not enabled/available)

exception IoticAgent.IOT.Exceptions.**IOTSyncTimeout** (*msg*, *event=None*)

Bases: *IoticAgent.IOT.Exceptions.IOTException*

Synchronous request timeout (see `IOT.sync_request_timeout` configuration option). Raised when a request does not complete within the timeout.

exception IoticAgent.IOT.Exceptions.**IOTUnknown** (*msg*, *event=None*)

Bases: *IoticAgent.IOT.Exceptions.IOTException*

Request FAILED because of Unknown resource (EG Thing not found)

4.1.4 IoticAgent.IOT.Point module

Wrapper object for Iotic Points.

Points are come in two types:

- Feed's where they output data from a Thing
- Control's where they are a way of sending data to a Thing

class IoticAgent.IOT.Point.**Control** (*client*, *lid*, *pid*, *guid*)

Bases: *IoticAgent.IOT.Point.Point*

Controls are where a Thing invites others to send it data. Controls can be used to activate some hardware, reset counters, change reporting intervals - pretty much anything you want to change the state of a Thing. Controls are many-to-one.

class IoticAgent.IOT.Point.**Feed** (*client, lid, pid, guid*)

Bases: *IoticAgent.IOT.Point.Point*

Feeds are advertised when a Thing has data to share. They are for out-going data which will get shared with any remote Things that have followed them. Feeds are one-to-many.

get_recent_info ()

Retrieves statistics and configuration about recent storage for this Feed.

Returns QAPI recent info function payload

```
{
    "maxSamples": 0,
    "count": 0
}
```

Raises

- *IoTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

get_template ()

Get new *IoticAgent.IOT.PointValueHelper module* *PointDataObject* instance to use for sharing data.

set_recent_config (*max_samples=0*)

Update/configure recent data settings for this Feed. If the container does not support recent storage or it is not enabled for this owner, this function will have no effect.

Parameters *max_samples* (*int, optional*) – How many shares to store for later retrieval. If not supported by container, this argument will be ignored. A value of zero disables this feature whilst a negative value requests the maximum sample store amount.

Returns QAPI recent config function payload

```
{
    "maxSamples": 0
}
```

Raises

- *IoTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

share (*data, mime=None, time=None*)

Share some data from this Feed

Raises

- *IoTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

Parameters

- **data** – The data you want to share
- **time** (*datetime*) – UTC time for this share. If not specified, the container's time will be used. Thus it makes almost no sense to specify *datetime.utcnow()* here. This parameter can be used to indicate that the share time does not correspond to the time to which the data applies, e.g. to populate recent storage with historical data.

- **mime** (*string*) – The mime type of the data you’re sharing. There are some Iotic Labs-defined default values.

“*idx/1*” corresponds to “application/ubjson” - the recommended way to send mixed data. Share a python dictionary as the data and the agent will do the encoding and decoding for you.

```
data = {}
data["temperature"] = self._convert_to_celsius(ADC.read(1))
# ...etc...
my_feed.share(data)
```

“*idx/2*” Corresponds to “text/plain” - the recommended way to send textual data. Share a utf8 string as data and the agent will pass it on, unchanged.

```
my_feed.share(u"string data")
```

“*text/xml*” or any other valid mime type. To show the recipients that you’re sending something more than just bytes

```
my_feed.share("<xml>...</xml>".encode('utf8'), mime="text/xml")
```

share_async (*data*, *mime=None*, *time=None*)

class IoticAgent.IOT.Point.**Point** (*client*, *lid*, *pid*, *guid*)

Bases: *IoticAgent.IOT.Resource.Resource*

create_tag (*tags*)

Create tags for a Point in the language you specify. Tags can only contain alphanumeric (unicode) characters and the underscore. Tags will be stored lower-cased.

Raises

- *IoTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

Parameters *tags* (*list*) – The list of tags you want to add to your Point, e.g. [“garden”, “soil”]

create_value (*label*, *vtype*, *lang=None*, *description=None*, *unit=None*)

Create a value on this Point. Values are descriptions in semantic metadata of the individual data items you are sharing (or expecting to receive, if this Point is a control). This will help others to search for your feed or control. If a value with the given label (and language) already exists, its fields are updated with the provided ones (or unset, if None).

Raises

- *IoTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

Parameters

- **label** (*string*) – The label for this value e.g. “Temperature”. The label must be unique for this Point. E.g. You can’t have two data values called “Volts” but you can have “volts1” and “volts2”.
- **lang** (*string*, *optional*) – The two-character ISO 639-1 language code to use for the description. None means use the default language for your agent. See *IoticAgent.IOT.Config module*

- **(xsd (vtype) – datatype)**: The datatype of the data you are describing, e.g. `dateTime`. We recommend you use a Iotic Labs-defined constant from *IoticAgent.Datatypes module*
- **description (string, optional)** – The longer descriptive text for this value.
- **unit (ontology url, optional)** – The url of the ontological description of the unit of your value. We recommend you use a constant from *IoticAgent.Units module*

```
# example with no units as time is unit-less
my_feed.create_value("timestamp",
                    Datatypes.DATETIME,
                    "en",
                    "time of reading")
```

```
# example with a unit from the Units class
my_feed.create_value("temperature",
                    Datatypes.DECIMAL,
                    "en",
                    "Fish-tank temperature in celsius",
                    Units.CELSIUS)
```

delete_tag (tags)

Delete tags for a Point in the language you specify. Case will be ignored and any tags matching lower-cased will be deleted.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters tags (list) – The list of tags you want to delete from your Point, e.g. [“garden”, “soil”]

delete_value (label=None)

Delete the labelled value (or all values) on this Point

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters label (string, optional) – The label for the value you want to delete. If not specified, all values for this point will be removed.

foc

Whether this Point is a feed or control. String of either “feed” or “control”

get_meta ()

Get the metadata object for this Point

Returns A *IoticAgent.IOT.PointMeta module* PointMeta object

OR**Raises**

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

get_meta_rdf (fmt='n3')

Get the metadata for this Point in rdf fmt

Advanced users who want to manipulate the RDF for this Point directly without the *IoticAgent.IOT.PointMeta module* PointMeta)helper object

Returns The RDF in the format you specify.

OR

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters `fmt` (*string, optional*) – The format of RDF you want returned. Valid formats are: “xml”, “n3”, “turtle”

lid

The local id of the Thing that advertises this Point. This is unique to you on this container.

list (*limit=50, offset=0*)

List *all* the values on this Point.

Returns QAPI list function payload

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **limit** (*integer, optional*) – Return this many value details
- **offset** (*integer, optional*) – Return value details starting at this offset

list_followers ()

List followers for this point, i.e. remote follows for feeds and remote attaches for controls.

Returns QAPI subscription list function payload

```
{
  "<Subscription GUID 1>": "<GUID of follower1>",
  "<Subscription GUID 2>": "<GUID of follower2>"
}
```

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **limit** (*integer, optional*) – Return this many value details
- **offset** (*integer, optional*) – Return value details starting at this offset

list_tag (*limit=50, offset=0*)

List *all* the tags for this Point

Returns List of tags, as below

```
[
    "mytag1",
    "mytag2"
    "ein_name",
    "nochein_name"
]
```

OR

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **limit** (*integer, optional*) – Return at most this many tags
- **offset** (*integer, optional*) – Return tags starting at this offset

pid

Point id - the local id of this Point. This is unique to you on this container. Think of it as a nickname for the Point

rename (*new_pid*)

Rename the Point.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters **new_pid** (*string*) – The new local identifier of your Point

set_meta_rdf (*rdf, fmt='n3'*)

Set the metadata for this Point in rdf fmt

class `IoticAgent.IOT.Point.PointDataObject` (*values, value_filter*)

Bases: `object`

Represents a point data reading or template for filling in values, ready to be e.g. shared. NOT threadsafe.

empty

True if no values have been set yet.

Type Returns

filter_by (*text=(), types=(), units=(), include_unset=False*)

Return subset of values which match the given text, types and/or units. For a value to be matched, at least one item from each specified filter category has to apply to a value. Each of the categories must be specified as a sequence of strings. If *include_unset* is set, unset values will also be considered.

missing

List of values which do not have a value set yet

to_dict ()

Converts the set of values into a dictionary. Unset values are excluded.

unset ()

Unsets all values

values

List of all values

4.1.5 IoticAgent.IOT.PointMeta module

Helper object for getting and setting metadata for an Iotic Point programmatically

class IoticAgent.IOT.PointMeta.**PointMeta** (*parent, rdf, default_lang, fmt='n3'*)

Bases: *IoticAgent.IOT.ResourceMeta.ResourceMeta*

PointMeta class does nothing more than its base class: ResourceMeta

4.1.6 IoticAgent.IOT.PointValueHelper module

Shareable object helper classes

class IoticAgent.IOT.PointValueHelper.**PointDataObjectHandler** (*point, client*)

Bases: object

Caches metadata for a point and produces skeletal representation of a point's values, optionally populating it with data from a share/ask/tell. Threadsafe.

get_template (*data=None*)

Get new template which represents the values of this point in a PointDataObject from the *IoticAgent.IOT.Point module*. If data is set (to a dictionary), use this to populate the created template.

exception IoticAgent.IOT.PointValueHelper.**RefreshException**

Bases: Exception

Raised by __refresh() to indicate metadata unsuitable for template usage.

class IoticAgent.IOT.PointValueHelper.**Value** (*label, type_, unit=None, description=None*)

Bases: object

Represent data and metadata for a single value. NOT thread safe.

copy ()

description

Human-readable description of this value

label

Label for this value

type_

Value type, e.g. one of IoticAgent.Datatypes

unit

Value unit, e.g. one of IoticAgent.Units

unset

Whether this value instances has had data assigned to it

value

Data for this value.

Returns None if it hasn't been set. To distinguish between None and and unset, check the *unset* property.

4.1.7 IoticAgent.IOT.RemotePoint module

class IoticAgent.IOT.RemotePoint.**RemoteControl** (*client, subid, pointid, lid*)

Bases: *IoticAgent.IOT.RemotePoint.RemotePoint*

Helper object for Control Subscription. A subscription the connection you have to another Thing's control. This object allows you to pass data to the other Things control in two ways, *ask* and *tell*:

ask where you "fire and forget" - the receiving object doesn't have to let you know whether it has actioned your request

tell where you expect the receiving object to let you know whether or not has actioned your request

When you attach to a Control this object will be returned from *IoticAgent.IOT.Thing module* Thing.attach.

ask (*data*, *mime=None*)

Request a remote control to do something. Ask is "fire-and-forget" in that you won't receive any notification of the success or otherwise of the action at the far end.

Raises

- *IOTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

Parameters

- **data** – The data you want to share
- **mime** (*string*, *optional*) – The mime type of the data you're sharing. See: *IoticAgent.IOT.Point module* Feed.share)

ask_async (*data*, *mime=None*)

get_template ()

Retreive *IoticAgent.IOT.PointValueHelper module* PointDataObject instance to use with this control.

tell (*data*, *timeout=10*, *mime=None*)

Order a remote control to do something. Tell is confirmed in that you will receive a notification of the success or otherwise of the action at the far end via a callback

Example

```
data = {"thermostat":18.0}
retval = r_thermostat.tell(data, timeout=10, mime=None)
if retval is not True:
    print("Thermostat not reset - reason: {reason}".format(reason=retval))
```

Returns

Return type True on success or else returns the reason (string) one of

```
"timeout"      # The request-specified timeout has been reached.
"unreachable"  # The remote control is not associated with an agent
               # or is not reachable in some other way.
"failed"       # The remote control indicates it did not perform
               # the request.
```

Raises

- *IOTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

Parameters

- **data** – The data you want to share

- **timeout** (*int*, *optional*) – Default 10. The delay in seconds before your tell request times out
- **mime** (*string*, *optional*) – See: *IoticAgent.IOT.Point module* Feed.share

tell_async (*data*, *timeout=10*, *mime=None*)

Asynchronous version of *IoticAgent.IOT.RemotePoint module* RemoteControl.tell.

Note: payload contains the success and reason keys they are not separated out as in the synchronous version

class IoticAgent.IOT.RemotePoint.**RemoteFeed** (*client*, *subid*, *pointid*, *lid*)

Bases: *IoticAgent.IOT.RemotePoint.RemotePoint*

Helper object for Feed Subscription.

A subscription the connection you have to another Thing's feed. This object allows you to simulate a feed being received by your Thing and also to return you the last known received feed data.

When you subscribe to a Feed this object will be returned from *IoticAgent.IOT.Thing module* Thing.follow. This helper object provides *simulate()* and *get_last()*

get_last ()

Returns None if no recent data is available for this point or a dict.

Dict contains:

```
'data' # (decoded or raw bytes)
'mime' # (None, unless payload was not decoded and has a mime type)
'time' # (datetime representing UTC timestamp of share)
```

Note: Shorthand for *get_recent(1)*.

get_recent (*count*)

Get the last instance(s) of feeddata from the feed. Useful if the remote Thing doesn't publish very often.

Returns An iterable of dicts (in chronologically ascending order) containing

```
'data' # (decoded or raw bytes)
'mime' # (None, unless payload was not decoded and has a mime type)
'time' # (datetime representing UTC timestamp of share)
```

Parameters **count** (*integer*) – How many recent instances to retrieve. High values might be floored to a maximum as defined by the container.

Note: Feed data is iterable as soon as it arrives, rather than when the request completes.

get_recent_async (*count*, *callback*)

Similar to *get_recent* except instead of returning an iterable, passes each dict to the given function which must accept a single argument.

Returns The request.

Parameters **callback** (*function*) – instead of returning an iterable, pass each dict (as described above) to the given function which must accept a single argument. Nothing is returned.

simulate (*data, mime=None*)

Simulate the arrival of feeddata into the feed. Useful if the remote Thing doesn't publish very often.

Parameters

- **data** – The data you want to use to simulate the arrival of remote feed data
- **mime** (*string, optional*) – The mime type of your data. See: [IoticAgent.IOT.Point module](#) `Feed.share`

class `IoticAgent.IOT.RemotePoint.RemotePoint` (*client, subid, pointid, lid*)

Bases: `object`

Base class for remote point types

guid

The Globally Unique ID of the Point you've followed (or attached to) in hex form (undashed).

lid

Local id of thing which is following to this feed / attached to this control

subid

Advanced users only The global subscription ID for the connection to this remote point in hex form (undashed).

4.1.8 IoticAgent.IOT.Resource module

Wrapper object for Iotic resources

class `IoticAgent.IOT.Resource.Resource` (*client, guid*)

Bases: `object`

Resource base class

guid

The Globally Unique ID of this resource in hex form (undashed).

4.1.9 IoticAgent.IOT.ResourceMeta module

Base class for Helper objects for getting and setting metadata programmatically

class `IoticAgent.IOT.ResourceMeta.ResourceMeta` (*parent, rdf, default_lang, fmt='n3'*)

Bases: `object`

Base class for metadata helpers. Inherited by:

- [IoticAgent.IOT.PointMeta module](#)
- [IoticAgent.IOT.ThingMeta module](#)

Do not instantiate directly

delete_description (*lang=None*)

Deletes all the *label* metadata properties on your Thing/Point for this language

Raises `ValueError` – Contains an error message if the parameters fail validation

Parameters `lang` (*string, optional*) – The two-character ISO 639-1 language code to identify your label. None means use the default language for your agent. See [IoticAgent.IOT.Config module](#)

delete_label (*lang=None*)

Deletes all the *label* metadata properties on your Thing/Point for this language

Raises `ValueError` – Contains an error message if the parameters fail validation

Parameters `lang` (*string, optional*) – The two-character ISO 639-1 language code to identify your label. None means use the default language for your agent. See [IoticAgent.IOT.Config module](#)

get_descriptions ()

Gets all the *description* metadata properties on your Thing/Point. Only one description is allowed per language, so you'll get a list of descriptions in the *N3* syntax, e.g. [*"fish tank"@en, "aquarium"@fr*]

Returns List of descriptions in N3 format or empty list if none.

get_descriptions_rdf ()

Gets all the *description* metadata properties on your Thing/Point. Only one description is allowed per language, so you'll get a list of descriptions as `rdflib.term.Literals`

Returns List of descriptions as `rdflib.term.Literals` or empty list if none.

get_labels ()

Gets all the *label* metadata properties on your Thing/Point. Only one label is allowed per language, so you'll get a list of labels in the *N3* syntax, e.g. [*"fish"@en, "poisson"@fr*]

Returns List of labels in N3 format

get_labels_rdf ()

Gets all the *label* metadata properties on your Thing/Point. Only one label is allowed per language, so you'll get a list of labels as `rdflib.term.Literal` objects

Returns List of labels as `rdflib.term.Literals`

set ()

Pushes the RDF metadata description back to the infrastructure. This will be searchable if you have called [IoticAgent.IOT.Thing module](#) `Thing.set_public` at any time.

Example 1 - Recommended

Use of python *with* syntax and `XXXXmeta` class.

```
# using with calls set() for you so you don't forget
with thing_solar_panels.get_meta() as meta_thing_solar_panels:
    meta_thing_solar_panels.set_label("Mark's Solar Panels")
    meta_thing_solar_panels.set_description("Solar Array 3.3kW")
    meta_thing_solar_panels.set_location(52.1965071,0.6067687)
```

Example 2

Explicit use of `set`

```
meta_thing_solar_panels = thing_solar_panels.get_meta()

meta_thing_solar_panels.set_label("Mark's Solar Panels")
meta_thing_solar_panels.set_description("Solar Array 3.3kW")
meta_thing_solar_panels.set_location(52.1965071,0.6067687)

meta_thing_solar_panels.set()
```

Raises

- `IoTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure
- `rdflib.plugins.parsers.notation3.BadSyntax` – if the RDF is badly formed *n3*
- `xml.sax._exceptions.SAXParseException` – if the RDF is badly formed *xml*

set_description (*description*, *lang=None*)

Sets the *description* metadata property on your Thing/Point. Only one description is allowed per language, so any other descriptions in this language are removed before adding this one

Raises `ValueError` – Contains an error message if the parameters fail validation

Parameters

- **description** (*string*) – the new text of the description
- **lang** (*string*, *optional*) – The two-character ISO 639-1 language code to identify your label. None means use the default language for your agent. See [IoticAgent.IOT.Config module](#)

set_label (*label*, *lang=None*)

Sets the *label* metadata property on your Thing/Point. Only one label is allowed per language, so any other labels in this language are removed before adding this one

Raises `ValueError` – Contains an error message if the parameters fail validation

Parameters

- **label** (*string*) – the new text of the label
- **lang** (*string*, *optional*) – The two-character ISO 639-1 language code to use for your label. None means use the default language for your agent. See [IoticAgent.IOT.Config module](#).

update ()

Gets the latest version of your metadata from the infrastructure and updates your local copy

Returns *True* if successful, *False* otherwise

OR**Raises**

- `IoTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

4.1.10 IoticAgent.IOT.Thing module

Wrapper object for Iotic Things

class `IoticAgent.IOT.Thing.Thing` (*client*, *lid*, *guid*, *epId*)

Bases: [IoticAgent.IOT.Resource.Resource](#)

Thing class

agent_id

Agent id (aka epId) with which this Thing is associated. None indicates this Thing is not assigned to any agent. The following actions can only be performed with a Thing if operating in its associated agent:

- Receive feed data from feeds the Thing is following

- Share feed data for feeds this Thing owns
- Receive control requests for controls the Thing owns
- Perform ask/tell on a control this Thing is attached to

Attempting to perform the above actions from another agent will result in either a local (e.g. `ValueError`) or remote exception to be raised(*IoticAgent.IOT.Exceptions module*).

attach (*gpId*)

Create a subscription (i.e. attach) to a Control-Point with a global point id (*gpId*) and a feed data callback

Returns A new `RemoteControl` object from the *IoticAgent.IOT.RemotePoint module* or the existing one if the subscription already exists

Raises

- `IoTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **gpId** (*lid*, *pid*) – Global id of the Point to which you want to attach **OR**
- **gpId** – Tuple of (*thing_localid*, *point_localid*) for local subscription

attach_async (*gpId*)

create_control (*pid*, *callback*, *callback_parsed=None*)

Create a control for this Thing with a local point id (*pid*) and a control request feedback

Returns A new *IoticAgent.IOT.Point module* control object or the existing one if the Control already exists

Raises

- `IoTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **pid** (*string*) – Local id of your Control
- **callback** (*function reference*) – Callback function to invoke on receipt of a control request.
- **callback_parsed** (*function reference*, *optional*) – Callback function to invoke on receipt of control data. This is equivalent to *callback* except the dict includes the *parsed* key which holds the set of values in a *IoticAgent.IOT.Point module* `PointDataObject` instance. If both *callback_parsed* and *callback* have been specified, the former takes precedence and *callback* is only called if the point data could not be parsed according to its current value description.

The *callback* receives a single dict argument, with keys of:

```
'data'      # (decoded or raw bytes)
'mime'      # (None, unless payload was not decoded and has a mime type)
'subId'     # (the global id of the associated subscription)
'entityLid' # (local id of the Thing to which the control belongs)
'lid'       # (local id of control)
'confirm'   # (whether a confirmation is expected)
'requestId' # (required for sending confirmation)
```

Note: `callback_parsed` can only be used if `auto_encode_decode` is enabled for the client instance.

create_control_async (*pid*, *callback*, *callback_parsed=None*)

create_feed (*pid*, *save_recent=0*)

Create a new Feed for this Thing with a local point id (*pid*).

Returns A new *IoticAgent.IOT.Point module* feed object, or the existing one, if the Feed already exists

Raises

- *IoTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

Parameters

- **pid** (*string*) – Local id of your Feed
- **save_recent** (*int*, *optional*) – How many shares to store for later retrieval. If not supported by container, this argument will be ignored. A value of zero disables this feature whilst a negative value requests the maximum sample store amount.

create_feed_async (*pid*, *save_recent=0*)

create_property (*props*, *replace=True*, *replace_all=False*)

Create RDF properties for a Thing, given as a list of predicate-object tuples. Strs, bools, ints, and floats will be automatically cast to the correct XSD type; any other type must be given as a third member of the tuple. Strings can be given a language with two @’s and the two-letter language code at the end, e.g. “Fritz@@de”. Iotic internal properties cannot be altered.

Raises

- *IoTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

Parameters

- **props** (*list*) – The sequence of properties (or single property) you want to add to your Thing
- **replace** (*bool*) – Whether to delete all existing properties for the given predicates before adding new ones.
- **replace_all** (*bool*) – Whether to delete all existing properties for ALL predicates before adding new ones.

create_tag (*tags*)

Create tags for a Thing in the language you specify. Tags can only contain alphanumeric (unicode) characters and the underscore. Tags will be stored lower-cased.

Raises

- *IoTException* – Infrastructure problem detected
- *LinkException* – Communications problem between you and the infrastructure

Parameters **tags** (*list*) – The list of tags (or single one) you want to add to your Thing, e.g. [“garden”, “soil”]

delete_control (*pid*)

Delete a control, identified by its local id.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters `pid` (*string*) – Local identifier of your control you want to delete

`delete_control_async` (*pid*)

`delete_feed` (*pid*)

Delete a feed, identified by its local id.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters `pid` (*string*) – Local identifier of your feed you want to delete

`delete_feed_async` (*pid*)

`delete_property` (*props*)

Delete RDF properties for a Thing. Properties should be given as a list of predicate-object tuples as shown in the `create_property` method; however, unlike that method, `None` can be given as an object, having the effect of deleting all the properties for that predicate. Also, an empty list can be passed, deleting all non-internal properties. Iotic internal properties cannot be altered.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters `properties` (*list*) – The sequence of properties (or single property) you want to delete from your Thing

`delete_tag` (*tags*)

Delete tags for a Thing in the language you specify. Case will be ignored and any tags matching lower-cased will be deleted.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters `tags` (*list*) – The list of tags (or single one) you want to delete from your Thing, e.g. [`"garden"`, `"soil"`]

`follow` (*gpid*, *callback=None*, *callback_parsed=None*)

Create a subscription (i.e. follow) a Feed/Point with a global point id (*gpid*) and a feed data callback

Returns A new *IoticAgent.IOT.RemotePoint module* RemoteFeed object or the existing one if the subscription already exists

Or:

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- `gpid` (*lid*, *pid*) – Global id of the Point you want to follow **OR**

- **gp_id** – Tuple of (*thing_localid*, *point_localid*) for local subscription
- **callback** (*function reference*, *optional*) – Callback function to invoke on receipt of feed data.
- **callback_parsed** (*function reference*, *optional*) – Callback function to invoke on receipt of feed data. This is equivalent to *callback* except the dict includes the *parsed* key which holds the set of values in a *loticAgent.IOT.Point module* *PointDataObject* instance. If both *callback_parsed* and *callback* have been specified, the former takes precedence and *callback* is only called if the point data could not be parsed according to its current value description.

Note: The callback receives a single dict argument, with keys of:

```
'data' # (decoded or raw bytes)
'mime' # (None, unless payload was not decoded and has a mime type)
'pid'  # (the global id of the feed from which the data originates)
'time' # (datetime representing UTC timestamp of share)
```

Note: *callback_parsed* can only be used if *auto_encode_decode* is enabled for the client instance.

follow_async (*gp_id*, *callback=None*, *callback_parsed=None*)

get_control (*pid*)

Get the details of a newly created control. This only applies to asynchronous creation of feeds and the new control instance can only be retrieved once.

Note: Destructive Read. Once you've called *get_control* once, any further calls will raise a *KeyError*

Returns A *loticAgent.IOT.Point module* control object, which corresponds to the cached entry for this local control id

Parameters **pid** (*string*) – Local identifier of your control.

Raises *KeyError* – The control has not been newly created (or has already been retrieved by a previous call)

get_feed (*pid*)

Get the details of a newly created feed. This only applies to asynchronous creation of feeds and the new feed instance can only be retrieved once.

Note: Destructive Read. Once you've called *get_feed* once, any further calls will raise a *KeyError*

Returns A *loticAgent.IOT.Point module* feed object, which corresponds to the cached entry for this local feed id.

Parameters **pid** (*string*) – Point id - local identifier of your feed.

Raises *KeyError* – Feed has not been newly created (or has already been retrieved by a previous call)

get_meta()

Get the metadata object for this Thing

Returns A *IoticAgent.IOT.ThingMeta module* object

get_meta_rdf(fmt='n3')

Get the metadata for this Thing in rdf fmt.

Advanced users who want to manipulate the RDF for this Thing directly without the *IoticAgent.IOT.ThingMeta module* helper object.

Returns The RDF in the format you specify

OR

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters **fmt** (*string, optional*) – The format of RDF you want returned. Valid formats are: “xml”, “n3”, “turtle”

get_remote_control(gpid)

Retrieve *RemoteControl* instance for an attach. This only applies to asynchronous attach requests and the new *RemoteControl* instance can only be retrieved once.

Note: Destructive Read. Once you’ve called `get_remote_control` once, any further calls will raise a *KeyError*

Raises

- `KeyError` – If the attach-subscription has not been newly created (or has already been retrieved by a previous call)
- `ValueError` – If the followed Point is actually a feed instead of a control, or if the subscription has not completed yet.

get_remote_feed(gpid)

Retrieve *RemoteFeed* instance for a follow. This only applies to asynchronous follow requests and the new *RemoteFeed* instance can only be retrieved once.

Note: Destructive Read. Once you’ve called `get_remote_feed` once, any further calls will raise a *KeyError*

Raises

- `KeyError` – The follow-subscription has not been newly created (or has already been retrieved by a previous call)
- `ValueError` – The followed Point is actually a control instead of a feed, or if the subscription has not completed yet.

lid

The local id of this Thing. This is unique to you on this container. Think of it as a nickname for the Thing

list_connections (*limit=500, offset=0*)

List Points to which this Things is subscribed. I.e. list all the Points this Thing is following and controls it's attached to

Returns Subscription list e.g.

```
{
  "<Subscription GUID 1>": {
    "id": "<Control GUID>",
    "entityId": "<Control's Thing GUID>",
    "type": 3 # R_CONTROL from IoticAgent.Core.Const
  },
  "<Subscription GUID 2>": {
    "id": "<Feed GUID>",
    "entityId": "<Feed's Thing GUID>",
    "type": 2 # R_FEED from IoticAgent.Core.Const
  }
}
```

Raises

- `IoTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Note: For Things following a Point see *IoticAgent.IOT.Point module* `list_followers`.

list_controls (*limit=500, offset=0*)

List *all* the controls on this Thing.

Returns QAPI list function payload

Raises

- `IoTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **limit** (*integer, optional*) – Return this many Point details
- **offset** (*integer, optional*) – Return Point details starting at this offset

list_feeds (*limit=500, offset=0*)

List *all* the feeds on this Thing.

Returns QAPI list function payload

Raises

- `IoTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **limit** (*integer, optional*) – Return this many Point details
- **offset** (*integer, optional*) – Return Point details starting at this offset

list_property (*limit=500, offset=0*)

List *all* the properties for this Thing

Returns dict with predicates as keys and object lists as values, with RDF Literals converted to the format given as input to `create_property` (ie, XSD type as extra tuple member where required, language via '@@en' suffix). Iotic-internal properties omitted. Eg:

```
{
  "http://xmlns.com/foaf/0.1/age": [
    31
  ],
  "http://my.domain.net/some/ontology#customPredicate": [
    55.4
  ],
  "http://my.domain.net/some/ontology#someFlag": [
    True
  ],
  "http://my.domain.net/some/ontology#someProperty": [
    "A string without language"
  ],
  "http://my.domain.net/some/ontology#anotherProperty2": [
    "Language-specific string@@en"
  ],
  "http://www.w3.org/1999/02/22-rdf-syntax-ns#type": [
    "http://some.company.org/my-types/oven",
    "http://some.company.org/my-types/hob"
  ],
  "http://my.domain.net/some/ontology#anotherProperty": [
    "The english text@@en",
    "Die deutsche Version@@de"
  ],
  "http://my.domain.net/some/ontology#buildMonth": [
    ("2019-11", "gYearMonth")
  ],
  "http://my.domain.net/some/ontology#purchaseTime": [
    ("2018-10-10T12:00:00Z", "dateTime"),
    ("2019-11-12T14:21:44Z", "dateTime")
  ]
}
```

Or:

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **limit** (*integer, optional*) – Return at most this many properties
- **offset** (*integer, optional*) – Return properties starting at this offset

list_tag (*limit=500, offset=0*)

List *all* the tags for this Thing

Returns Lists of tags, as below

```
[
  "mytag1",
  "mytag2"
  "ein_name",
```

(continues on next page)

(continued from previous page)

```

    "nochein_name"
]

```

Or:

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **limit** (*integer, optional*) – Return at most this many tags
- **offset** (*integer, optional*) – Return tags starting at this offset

reassign (*new_epid*)

Reassign the Thing from one agent to another.

Advanced users only

This will lead to any local instances of a Thing being rendered useless. They won't be able to receive control requests, feed data or to share any feeds as they won't be in this agent.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters **new_epid** (*string*) – The new agent id to which your Thing should be assigned.

If None, current agent will be chosen. If False, existing agent will be unassigned.

rename (*new_lid*)

Rename the Thing.

Advanced users only

This can be confusing. You are changing the local id of a Thing to *new_lid*. If you create another Thing using the "old_lid", the system will oblige, but it will be a completely **new** Thing.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters **new_lid** (*string*) – The new local identifier of your Thing

set_meta_rdf (*rdf, fmt='n3'*)

Set the metadata for this Thing in RDF fmt.

Advanced users who want to manipulate the RDF for this Thing directly without the *IoticAgent.IOT.ThingMeta* module helper object.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters **fmt** (*string, optional*) – The format of RDF you have sent. Valid formats are: "xml", "n3", "turtle"

set_public (*public=True*)

Sets your Thing to be public to all if *public=True*. This means the tags, label and description of your Thing are now searchable by anybody, along with its location and the units of any values on any Points.

If *public=False* the metadata of your Thing is no longer searchable.

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters **public** (*boolean, optional*) – Whether (or not) to allow your Thing's metadata to be searched by anybody

unattach (*subid*)

Remove a subscription of a control with a global subscription id (gpId)

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **subid** (*object*) – Global id of the subscription you want to delete **OR**
- **subid** – The instance of a RemoteControl object corresponding to the control you want to cease being able to actuate.

unfollow (*subid*)

Remove a subscription of a Feed with a global subscription id (gpId)

Raises

- `IOTException` – Infrastructure problem detected
- `LinkException` – Communications problem between you and the infrastructure

Parameters

- **subid** (*object*) – Global id of the subscription you want to delete **OR**
- **subid** – The instance of a RemoteFeed object corresponding to the feed you want to cease following.

4.1.11 IoticAgent.IOT.ThingMeta module

Helper object for getting and setting metadata for Iotic Things programmatically

class `IoticAgent.IOT.ThingMeta.ThingMeta` (*parent, rdf, default_lang, fmt='n3'*)

Bases: `IoticAgent.IOT.ResourceMeta.ResourceMeta`

Metadata helper for Thing class. Wraps RDF graph and provides simple get/set/delete methods for location. All other methods inherited from ResourceMeta

delete_location ()

Deletes all the *geo:lat* and *geo:long* metadata properties on your Thing

get_location ()

Gets the current geo location of your Thing

Returns Tuple of (*lat, lon*) in *float* or (*None, None*) if location is not set for this Thing

set_location (*lat, lon*)

4.1.12 IoticAgent.IOT.utils module

Utility functions

`IoticAgent.IOT.utils.bool_from(obj, default=False)`

Returns True if obj is not None and its string representation is not 0 or False (case-insensitive). If obj is None, 'default' is used.

`IoticAgent.IOT.utils.foc_to_str(foc)`

Take the QAPI Resource code and return string of 'feed' or 'control'

`IoticAgent.IOT.utils.hex_to_uuid(hstr)`

Make a 32 ascii char hex into a 8-4-4-4-12 uuid

`IoticAgent.IOT.utils.private_name_for(var_name, cls)`

Returns Mangled variable name (if applicable) for the given variable and class instance. See <https://docs.python.org/3/tutorial/classes.html#private-variables>

`IoticAgent.IOT.utils.private_names_for(cls, names)`

Returns Iterable of private names using privateNameFor()

`IoticAgent.IOT.utils.str_to_foc(sfoc)`

Take a string of 'feed' or 'control' and return the QAPI Resource Const.R_FEED or R_CONTROL

`IoticAgent.IOT.utils.uuid_to_hex(uid)`

Make 8-4-4-4-12 uuid into plain hex for the QAPI

`IoticAgent.IOT.utils.version_string_to_tuple(version)`

IoticAgent.ThingRunner module

class IoticAgent.ThingRunner.**RetryingThingRunner** (*config=None, retry_timeout=15*)

Bases: *IoticAgent.ThingRunner.ThingRunner*

Automatically re-tries on_startup & on_main on network & timeout related failures only.

on_exception (*ctx, exc_info*)

Called when an exception occurs within runner methods (or initialisation). If the return value evaluates to True, the method in question will be re-tried (after *retry_timeout* seconds wait). Otherwise the exception will be re-raised (the default). Note that KeyboardInterrupt will not result in this method being called and instead cause a shutdown.

ctx One of *RunContext*. Indicates at what point exception occurred. *exc_info* Tuple (as for *sys.exc_info()*) of the exception

class IoticAgent.ThingRunner.**RunContext**

Bases: *enum.Enum*

Passed to *ThingRunner.on_exception*

MAIN = 'main'

ON_STARTUP = 'on_startup'

class IoticAgent.ThingRunner.**ThingRunner** (*config=None, retry_timeout=15*)

Bases: *object*

Automates, starting, stopping and running of an Agent instance, either in the foreground (blocking) or background. Create a subclass to use, e.g.:

```
class MyRunner(ThingRunner):
    # only required if want to add own fields to class instance
    def __init__(self, other, arguments, config=None):
        super(ThingRunner, self).__init__(config=config)
        # own class setup goes here

    def main(self):
        while True:
```

(continues on next page)

(continued from previous page)

```

        # do stuff here

        # end on shutdown request
        if self.wait_for_shutdown(1):
            break

# runs in foreground, blocking
MyRunner('other', 'arguments', config='agent.ini').run()

```

Optionally implement *on_startup* or *on_shutdown* to perform one-off actions at the beginning/end of the agent's run cycle.

client

IoticAgent.IOT.Client module instance in use by this runner

main()

Application logic goes here. Should return (or raise exception) to end program run. Should check whether the *shutdown_requested* property is True and return if this is the case.

on_exception (ctx, exc_info)

Called when an exception occurs within runner methods (or initialisation). If the return value evaluates to True, the method in question will be re-tried (after *retry_timeout* seconds wait). Otherwise the exception will be re-raised (the default). Note that KeyboardInterrupt will not result in this method being called and instead cause a shutdown.

ctx One of *RunContext*. Indicates at what point exception occurred. *exc_info* Tuple (as for *sys.exc_info()*) of the exception

on_shutdown (exc_info)

One-off tasks to perform on just before agent shutdown. *exc_info* is a tuple (as for *sys.exc_info()*) of the exception which caused the shutdown (from the *main()* function) or None if the shutdown was graceful. This is useful if one only wants to perform certain tasks on success. This is not called if *on_startup()* was not successful. It is possible that due to e.g. network problems the agent cannot be used at this point. If the return value evaluates to False, the exception will be re-raised (the default). Note that KeyboardInterrupt will not be passed to this method (but will result in this method being called).

on_startup ()

One-off tasks to perform straight **after** agent startup.

run (background=False)

Runs *on_startup*, *main* and *on_shutdown*, blocking until finished, unless background is set.

shutdown_requested

Whether *stop()* has been called, an exception has occurred (which does not result in a retry) or the implemented main loop has finished and thus the device should be shutting down.

stop (timeout=None)

Requests device to stop running, waiting at most the given timeout in seconds (fractional). Has no effect if *run()* was not called with *background=True* set.

Returns True if successfully stopped (or already not running).

wait_for_shutdown (timeout=None)

Blocks until shutdown has been requested (or the timeout has been reached, if specified). False is returned for the latter, True otherwise.

IoticAgent.Units module

Constants for Units and helper class to build complete JSON tree of units from

`IoticAgent.Units.AMPERE = 'http://purl.obolibrary.org/obo/UO_0000011'`
 Electric current unit

`IoticAgent.Units.BECQUEREL = 'http://purl.obolibrary.org/obo/UO_0000132'`
 An activity (of a radionuclide) unit which is equal to the activity of a quantity of radioactive material in which one nucleus decays per second or there is one atom disintegration per second

`IoticAgent.Units.CELSIUS = 'http://purl.obolibrary.org/obo/UO_0000027'`
 Temperature units where the freezing point of water at 273.15 Kelvin is considered 0C and the boiling point 283.15K is 100C

`IoticAgent.Units.CENTIMETER = 'http://purl.obolibrary.org/obo/UO_0000015'`
 Metric distance unit = 1/100 of a meter

`IoticAgent.Units.COUNTS_PER_MIN = 'http://purl.obolibrary.org/obo/UO_0000148'`
 An activity (of a radionuclide) unit which is equal to the number of light emissions produced by ionizing radiation in one minute.

`IoticAgent.Units.DECIBEL = 'http://purl.obolibrary.org/obo/UO_0000259'`
 A ratio unit which is an indicator of sound power per unit area.

`IoticAgent.Units.DEGREE = 'http://purl.obolibrary.org/obo/UO_0000185'`
 A plane angle unit which is equal to 1/360 of a full rotation or 1.7453310^{-2} rad

`IoticAgent.Units.FAHRENHEIT = 'http://purl.obolibrary.org/obo/UO_0000195'`
 A temperature unit which is equal to 5/9ths of a kelvin. Negative 40 degrees Fahrenheit is equal to negative 40 degrees Celsius

`IoticAgent.Units.GRAM = 'http://purl.obolibrary.org/obo/UO_0000021'`
 A mass unit which is equal to one thousandth of a kilogram or 10^{-3} kg

`IoticAgent.Units.HERTZ = 'http://purl.obolibrary.org/obo/UO_0000106'`
 A frequency unit which is equal to 1 complete cycle of a recurring phenomenon in 1 second.

`IoticAgent.Units.HOUR = 'http://purl.obolibrary.org/obo/UO_0000032'`
 A time unit which is equal to 3600 seconds or 60 minutes

`IoticAgent.Units.JOULE = 'http://purl.obolibrary.org/obo/VO_0000112'`
An energy unit which is equal to the energy required when a force of 1 newton moves an object 1 meter

`IoticAgent.Units.KELVIN = 'http://purl.obolibrary.org/obo/VO_0000012'`
A thermodynamic temperature unit. 0K is “absolute zero”, ~293K is “room temperature”, i.e. 20C

`IoticAgent.Units.KILOGRAM = 'http://purl.obolibrary.org/obo/VO_0000009'`
A mass unit which is equal to the mass of the International Prototype Kilogram kept by the BIPM at Sèvres, France.

`IoticAgent.Units.KILOWATT_HOUR = 'http://purl.obolibrary.org/obo/VO_0000224'`
An energy unit which is equal to 1000 Watt-hours

`IoticAgent.Units.LUX = 'http://purl.obolibrary.org/obo/VO_0000116'`
An illuminance unit which is equal to the illuminance produced by 1 lumen evenly spread over an area 1 m²

`IoticAgent.Units.MEGAHERTZ = 'http://purl.obolibrary.org/obo/VO_0000325'`
A frequency unit which is equal 1 Million Hz

`IoticAgent.Units.METER = 'http://purl.obolibrary.org/obo/VO_0000008'`
Metric distance unit = 100 cm

`IoticAgent.Units.METER_PER_SEC = 'http://purl.obolibrary.org/obo/VO_0000094'`
A speed/velocity unit which is equal to the speed of an object traveling 1 meter distance in one second

`IoticAgent.Units.MICROGRAM = 'http://purl.obolibrary.org/obo/VO_0000023'`
A mass unit which is equal to one millionth of a gram or 10⁻⁶ g

`IoticAgent.Units.MILLIGRAM = 'http://purl.obolibrary.org/obo/VO_0000022'`
A mass unit which is equal to one thousandth of a gram or 10⁻³ g

`IoticAgent.Units.MILLIMETER = 'http://purl.obolibrary.org/obo/VO_0000016'`
Metric distance unit = 1/100 of a meter

`IoticAgent.Units.MINUTE = 'http://purl.obolibrary.org/obo/VO_0000031'`
A time unit which is equal to 60 seconds

`IoticAgent.Units.MM_MERCURY = 'http://purl.obolibrary.org/obo/VO_0000272'`
A unit of pressure equal to the amount of fluid pressure one millimeter deep in mercury at 0C

`IoticAgent.Units.NEWTON = 'http://purl.obolibrary.org/obo/VO_0000108'`
A force unit which is equal to the force required to cause an acceleration of 1m/s² of a mass of 1 Kg

`IoticAgent.Units.PARTS_PER_BILLION = 'http://purl.obolibrary.org/obo/VO_0000170'`
A dimensionless concentration notation which denotes the amount of a given substance in a total amount of 10⁹ regardless of the units of measure as long as they are the same.

`IoticAgent.Units.PARTS_PER_HUNDRED = 'http://purl.obolibrary.org/obo/VO_0000167'`
A dimensionless concentration notation which denotes the amount of a given substance in a total amount of 100 regardless of the units of measure as long as they are the same.

`IoticAgent.Units.PARTS_PER_MILLION = 'http://purl.obolibrary.org/obo/VO_0000169'`
A dimensionless concentration notation which denotes the amount of a given substance in a total amount of 1,000,000 regardless of the units of measure as long as they are the same.

`IoticAgent.Units.PARTS_PER_QUADRILLION = 'http://purl.obolibrary.org/obo/VO_0000172'`
A dimensionless concentration notation which denotes the amount of a given substance in a total amount of 10¹⁵ regardless of the units of measure as long as they are the same.

`IoticAgent.Units.PARTS_PER_THOUSAND = 'http://purl.obolibrary.org/obo/VO_0000168'`
A dimensionless concentration notation which denotes the amount of a given substance in a total amount of 1000 regardless of the units of measure as long as they are the same.

```

IoticAgent.Units.PARTS_PER_TRILLION = 'http://purl.obolibrary.org/obo/VO_0000171'
    A dimensionless concentration notation which denotes the amount of a given substance in a total amount of
    1012 regardless of the units of measure as long as they are the same.

IoticAgent.Units.PASCAL = 'http://purl.obolibrary.org/obo/VO_0000110'
    A pressure unit which is equal to the pressure or stress on a surface caused by a force of 1 newton spread over a
    surface of 1 m2

IoticAgent.Units.PERCENT = 'http://purl.obolibrary.org/obo/VO_0000187'
    A dimensionless ratio unit which denotes numbers as fractions of 100.

IoticAgent.Units.RADIAN = 'http://purl.obolibrary.org/obo/VO_0000123'
    A plane angle unit which is equal to the angle subtended at the center of a circle by an arc equal in length to the
    radius of the circle, approximately 57 degrees 17 minutes and 44.6 seconds.

IoticAgent.Units.ROENTGEN = 'http://purl.obolibrary.org/obo/VO_0000136'
    An exposure unit which is equal to the amount of radiation required to liberate positive and negative charges of
    one electrostatic unit of charge in 1 cm3 of air

IoticAgent.Units.SECOND = 'http://purl.obolibrary.org/obo/VO_0000010'
    A time unit which is equal to the duration of 9,192,631,770 periods of the radiation corresponding to the transi-
    tion between the two hyperfine levels of the ground state of the caesium 133 atom

IoticAgent.Units.SQUARE_CENTIMETER = 'http://purl.obolibrary.org/obo/VO_0000081'
    An area unit which is equal to an area enclosed by a square with sides each 1 centimeter long.

IoticAgent.Units.SQUARE_METER = 'http://purl.obolibrary.org/obo/VO_0000080'
    An area unit which is equal to an area enclosed by a square with sides each 1 meter long.

IoticAgent.Units.SQUARE_MILLIMETER = 'http://purl.obolibrary.org/obo/VO_0000082'
    An area unit which is equal to an area enclosed by a square with sides each 1 millimeter long.

class IoticAgent.Units.Units(uofn=None)
    Bases: object

    print_units (parent=None, indent=0, indentsize=4)

    save_json (jsonfn=None, pretty=True)
        Write a .json file with the units tree jsonfn='path/file.name' default os.getcwd() + 'units.json' pretty=True
        use JSON dumps pretty print for human readability

IoticAgent.Units.VOLT = 'http://purl.obolibrary.org/obo/VO_0000218'
    An electric potential difference unit which is equal to the work per unit charge

IoticAgent.Units.WATT = 'http://purl.obolibrary.org/obo/VO_0000114'
    A power unit which is equal to the power used when work is done at the rate of 1 joule per second

IoticAgent.Units.WATT_HOUR = 'http://purl.obolibrary.org/obo/VO_0000223'
    An energy unit which is equal to the amount of electrical energy equivalent to a one-watt load drawing power
    for one hour

```


CHAPTER 7

Indices and tables

- `genindex`
- `modindex`
- `search`

i

- `IoticAgent`, [1](#)
- `IoticAgent.Core`, [9](#)
- `IoticAgent.Core.AmqpLink`, [9](#)
- `IoticAgent.Core.Const`, [10](#)
- `IoticAgent.Core.Validation`, [10](#)
- `IoticAgent.Datatypes`, [13](#)
- `IoticAgent.IOT`, [15](#)
- `IoticAgent.IOT.Client`, [15](#)
- `IoticAgent.IOT.Config`, [28](#)
- `IoticAgent.IOT.Exceptions`, [29](#)
- `IoticAgent.IOT.Point`, [29](#)
- `IoticAgent.IOT.PointMeta`, [35](#)
- `IoticAgent.IOT.PointValueHelper`, [35](#)
- `IoticAgent.IOT.RemotePoint`, [35](#)
- `IoticAgent.IOT.Resource`, [38](#)
- `IoticAgent.IOT.ResourceMeta`, [38](#)
- `IoticAgent.IOT.Thing`, [40](#)
- `IoticAgent.IOT.ThingMeta`, [49](#)
- `IoticAgent.IOT.utils`, [50](#)
- `IoticAgent.ThingRunner`, [51](#)
- `IoticAgent.Units`, [53](#)

A

agent_id (*IoticAgent.IOT.Client.Client* attribute), 15
 agent_id (*IoticAgent.IOT.Thing.Thing* attribute), 40
 AMPERE (in module *IoticAgent.Units*), 53
 AmqpLink (class in *IoticAgent.Core.AmqpLink*), 9
 ask() (*IoticAgent.IOT.RemotePoint.RemoteControl* method), 36
 ask_async() (*IoticAgent.IOT.RemotePoint.RemoteControl* method), 36
 attach() (*IoticAgent.IOT.Thing.Thing* method), 41
 attach_async() (*IoticAgent.IOT.Thing.Thing* method), 41
 AUTO (*IoticAgent.Core.Const.DescribeScope* attribute), 10

B

BASE64 (in module *IoticAgent.Datatypes*), 13
 BECQUEREL (in module *IoticAgent.Units*), 53
 bool_check_convert() (*IoticAgent.Core.Validation.Validation* static method), 10
 bool_from() (in module *IoticAgent.IOT.utils*), 50
 BOOLEAN (in module *IoticAgent.Datatypes*), 13
 BYTE (in module *IoticAgent.Datatypes*), 13

C

callable_check() (*IoticAgent.Core.Validation.Validation* static method), 11
 CELSIUS (in module *IoticAgent.Units*), 53
 CENTIMETER (in module *IoticAgent.Units*), 53
 check_convert_string() (*IoticAgent.Core.Validation.Validation* static method), 11
 Client (class in *IoticAgent.IOT.Client*), 15
 client (*IoticAgent.ThingRunner.ThingRunner* attribute), 52

comment_check_convert() (*IoticAgent.Core.Validation.Validation* class method), 11
 Config (class in *IoticAgent.IOT.Config*), 28
 confirm_tell() (*IoticAgent.IOT.Client.Client* method), 15
 Control (class in *IoticAgent.IOT.Point*), 29
 copy() (*IoticAgent.IOT.PointValueHelper.Value* method), 35
 COUNTS_PER_MIN (in module *IoticAgent.Units*), 53
 create_control() (*IoticAgent.IOT.Thing.Thing* method), 41
 create_control_async() (*IoticAgent.IOT.Thing.Thing* method), 42
 create_feed() (*IoticAgent.IOT.Thing.Thing* method), 42
 create_feed_async() (*IoticAgent.IOT.Thing.Thing* method), 42
 create_property() (*IoticAgent.IOT.Thing.Thing* method), 42
 create_tag() (*IoticAgent.IOT.Point.Point* method), 31
 create_tag() (*IoticAgent.IOT.Thing.Thing* method), 42
 create_thing() (*IoticAgent.IOT.Client.Client* method), 16
 create_thing_async() (*IoticAgent.IOT.Client.Client* method), 16
 create_value() (*IoticAgent.IOT.Point.Point* method), 31

D

DATE (in module *IoticAgent.Datatypes*), 13
 DATETIME (in module *IoticAgent.Datatypes*), 13
 datetime_check_convert() (*IoticAgent.Core.Validation.Validation* class method), 11
 DECIBEL (in module *IoticAgent.Units*), 53
 DECIMAL (in module *IoticAgent.Datatypes*), 13

`default_lang` (*IoticAgent.IOT.Client.Client* attribute), 16
`DEGREE` (in module *IoticAgent.Units*), 53
`delete_control()` (*IoticAgent.IOT.Thing.Thing* method), 42
`delete_control_async()` (*IoticAgent.IOT.Thing.Thing* method), 43
`delete_description()` (*IoticAgent.IOT.ResourceMeta.ResourceMeta* method), 38
`delete_feed()` (*IoticAgent.IOT.Thing.Thing* method), 43
`delete_feed_async()` (*IoticAgent.IOT.Thing.Thing* method), 43
`delete_label()` (*IoticAgent.IOT.ResourceMeta.ResourceMeta* method), 39
`delete_location()` (*IoticAgent.IOT.ThingMeta.ThingMeta* method), 49
`delete_property()` (*IoticAgent.IOT.Thing.Thing* method), 43
`delete_tag()` (*IoticAgent.IOT.Point.Point* method), 32
`delete_tag()` (*IoticAgent.IOT.Thing.Thing* method), 43
`delete_thing()` (*IoticAgent.IOT.Client.Client* method), 16
`delete_thing_async()` (*IoticAgent.IOT.Client.Client* method), 16
`delete_value()` (*IoticAgent.IOT.Point.Point* method), 32
`describe()` (*IoticAgent.IOT.Client.Client* method), 16
`describe_scope_check_convert()` (*IoticAgent.Core.Validation.Validation* class method), 11
`DescribeScope` (class in *IoticAgent.Core.Const*), 10
`description` (*IoticAgent.IOT.PointValueHelper.Value* attribute), 35
`description_check_convert()` (*IoticAgent.Core.Validation.Validation* class method), 11
`dict_to_cp()` (in module *IoticAgent.IOT.Config*), 28
`DOUBLE` (in module *IoticAgent.Datatypes*), 13

E

`empty` (*IoticAgent.IOT.Point.PointDataObject* attribute), 34
`event` (*IoticAgent.IOT.Exceptions.IOTException* attribute), 29

F

`FAHRENHEIT` (in module *IoticAgent.Units*), 53
`Feed` (class in *IoticAgent.IOT.Point*), 29

`filter_by()` (*IoticAgent.IOT.Point.PointDataObject* method), 34
`FLOAT` (in module *IoticAgent.Datatypes*), 14
`foc` (*IoticAgent.IOT.Point.Point* attribute), 32
`foc_check()` (*IoticAgent.Core.Validation.Validation* static method), 11
`foc_to_str()` (in module *IoticAgent.IOT.utils*), 50
`follow()` (*IoticAgent.IOT.Thing.Thing* method), 43
`follow_async()` (*IoticAgent.IOT.Thing.Thing* method), 44
`FULL` (*IoticAgent.Core.Const.SearchType* attribute), 10

G

`get()` (*IoticAgent.IOT.Config.Config* method), 28
`get_control()` (*IoticAgent.IOT.Thing.Thing* method), 44
`get_descriptions()` (*IoticAgent.IOT.ResourceMeta.ResourceMeta* method), 39
`get_descriptions_rdf()` (*IoticAgent.IOT.ResourceMeta.ResourceMeta* method), 39
`get_feed()` (*IoticAgent.IOT.Thing.Thing* method), 44
`get_labels()` (*IoticAgent.IOT.ResourceMeta.ResourceMeta* method), 39
`get_labels_rdf()` (*IoticAgent.IOT.ResourceMeta.ResourceMeta* method), 39
`get_last()` (*IoticAgent.IOT.RemotePoint.RemoteFeed* method), 37
`get_location()` (*IoticAgent.IOT.ThingMeta.ThingMeta* method), 49
`get_meta()` (*IoticAgent.IOT.Point.Point* method), 32
`get_meta()` (*IoticAgent.IOT.Thing.Thing* method), 44
`get_meta_rdf()` (*IoticAgent.IOT.Point.Point* method), 32
`get_meta_rdf()` (*IoticAgent.IOT.Thing.Thing* method), 45
`get_recent()` (*IoticAgent.IOT.RemotePoint.RemoteFeed* method), 37
`get_recent_async()` (*IoticAgent.IOT.RemotePoint.RemoteFeed* method), 37
`get_recent_info()` (*IoticAgent.IOT.Point.Feed* method), 30
`get_remote_control()` (*IoticAgent.IOT.Thing.Thing* method), 45
`get_remote_feed()` (*IoticAgent.IOT.Thing.Thing* method), 45
`get_template()` (*IoticAgent.IOT.Point.Feed* method), 30

- get_template() (IoticAgent.IOT.PointValueHelper.PointDataObjectHandler method), 35
- get_template() (IoticAgent.IOT.RemotePoint.RemoteControl method), 36
- get_thing() (IoticAgent.IOT.Client.Client method), 17
- GRAM (in module IoticAgent.Units), 53
- guid (IoticAgent.IOT.RemotePoint.RemotePoint attribute), 38
- guid (IoticAgent.IOT.Resource.Resource attribute), 38
- guid_check_convert() (IoticAgent.Core.Validation.Validation static method), 11
- ## H
- HERTZ (in module IoticAgent.Units), 53
- hex_to_uuid() (in module IoticAgent.IOT.utils), 50
- HOURL (in module IoticAgent.Units), 53
- ## I
- INT (in module IoticAgent.Datatypes), 14
- INTEGER (in module IoticAgent.Datatypes), 14
- IOTAccessDenied, 29
- IOTClientError, 29
- IOTException, 29
- IoticAgent (module), 1
- IoticAgent.Core (module), 9
- IoticAgent.Core.AmqpLink (module), 9
- IoticAgent.Core.Const (module), 10
- IoticAgent.Core.Validation (module), 10
- IoticAgent.Datatypes (module), 13
- IoticAgent.IOT (module), 15
- IoticAgent.IOT.Client (module), 15
- IoticAgent.IOT.Config (module), 28
- IoticAgent.IOT.Exceptions (module), 29
- IoticAgent.IOT.Point (module), 29
- IoticAgent.IOT.PointMeta (module), 35
- IoticAgent.IOT.PointValueHelper (module), 35
- IoticAgent.IOT.RemotePoint (module), 35
- IoticAgent.IOT.Resource (module), 38
- IoticAgent.IOT.ResourceMeta (module), 38
- IoticAgent.IOT.Thing (module), 40
- IoticAgent.IOT.ThingMeta (module), 49
- IoticAgent.IOT.utils (module), 50
- IoticAgent.ThingRunner (module), 51
- IoticAgent.Units (module), 53
- IOTInternalError, 29
- IOTMalformed, 29
- IOTNotAllowed, 29
- IOTSyncTimeout, 29
- IOTUnknown, 29
- IRI (in module IoticAgent.Datatypes), 14
- is_alive() (IoticAgent.Core.AmqpLink.AmqpLink method), 9
- is_connected() (IoticAgent.IOT.Client.Client method), 18
- ## J
- JOULE (in module IoticAgent.Units), 54
- ## K
- KELVIN (in module IoticAgent.Units), 54
- KILOGRAM (in module IoticAgent.Units), 54
- KILOWATT_HOUR (in module IoticAgent.Units), 54
- ## L
- label (IoticAgent.IOT.PointValueHelper.Value attribute), 35
- label_check_convert() (IoticAgent.Core.Validation.Validation class method), 11
- lang_check_convert() (IoticAgent.Core.Validation.Validation class method), 11
- last_send_exc_time (IoticAgent.Core.AmqpLink.AmqpLink attribute), 9
- lid (IoticAgent.IOT.Point.Point attribute), 33
- lid (IoticAgent.IOT.RemotePoint.RemotePoint attribute), 38
- lid (IoticAgent.IOT.Thing.Thing attribute), 45
- lid_check_convert() (IoticAgent.Core.Validation.Validation class method), 11
- limit_offset_check() (IoticAgent.Core.Validation.Validation static method), 11
- list() (IoticAgent.IOT.Client.Client method), 18
- list() (IoticAgent.IOT.Point.Point method), 33
- list_connections() (IoticAgent.IOT.Thing.Thing method), 45
- list_controls() (IoticAgent.IOT.Thing.Thing method), 46
- list_feeds() (IoticAgent.IOT.Thing.Thing method), 46
- list_followers() (IoticAgent.IOT.Point.Point method), 33
- list_property() (IoticAgent.IOT.Thing.Thing method), 46
- list_tag() (IoticAgent.IOT.Point.Point method), 33
- list_tag() (IoticAgent.IOT.Thing.Thing method), 47
- LOCAL (IoticAgent.Core.Const.DescribeScope attribute), 10
- LOCAL (IoticAgent.Core.Const.SearchScope attribute), 10

local_meta (*IoticAgent.IOT.Client.Client* attribute), 18
 LOCAL_OWN (*IoticAgent.Core.Const.DescribeScope* attribute), 10
 LOCAL_OWN (*IoticAgent.Core.Const.SearchScope* attribute), 10
 LOCATED (*IoticAgent.Core.Const.SearchType* attribute), 10
 location_check() (*IoticAgent.Core.Validation.Validation* static method), 11
 LONG (in module *IoticAgent.Datatypes*), 14
 LUX (in module *IoticAgent.Units*), 54

M

MAIN (*IoticAgent.ThingRunner.RunContext* attribute), 51
 main() (*IoticAgent.ThingRunner.ThingRunner* method), 52
 MEGAHERTZ (in module *IoticAgent.Units*), 54
 metafmt_check_convert() (*IoticAgent.Core.Validation.Validation* static method), 11
 METER (in module *IoticAgent.Units*), 54
 METER_PER_SEC (in module *IoticAgent.Units*), 54
 MICROGRAM (in module *IoticAgent.Units*), 54
 MILLIGRAM (in module *IoticAgent.Units*), 54
 MILLIMETER (in module *IoticAgent.Units*), 54
 mime_check_convert() (*IoticAgent.Core.Validation.Validation* class method), 11
 MINIMAL (*IoticAgent.Core.Const.SearchType* attribute), 10
 MINUTE (in module *IoticAgent.Units*), 54
 missing (*IoticAgent.IOT.Point.PointDataObject* attribute), 34
 MM_MERCURY (in module *IoticAgent.Units*), 54

N

NEWTON (in module *IoticAgent.Units*), 54

O

object_type_check_convert() (*IoticAgent.Core.Validation.Validation* static method), 11
 on_exception() (*IoticAgent.ThingRunner.RetryingThingRunner* method), 51
 on_exception() (*IoticAgent.ThingRunner.ThingRunner* method), 52
 on_shutdown() (*IoticAgent.ThingRunner.ThingRunner* method), 52

ON_STARTUP (*IoticAgent.ThingRunner.RunContext* attribute), 51
 on_startup() (*IoticAgent.ThingRunner.ThingRunner* method), 52

P

PARTS_PER_BILLION (in module *IoticAgent.Units*), 54
 PARTS_PER_HUNDRED (in module *IoticAgent.Units*), 54
 PARTS_PER_MILLION (in module *IoticAgent.Units*), 54
 PARTS_PER_QUADRILLION (in module *IoticAgent.Units*), 54
 PARTS_PER_THOUSAND (in module *IoticAgent.Units*), 54
 PARTS_PER_TRILLION (in module *IoticAgent.Units*), 54
 PASCAL (in module *IoticAgent.Units*), 55
 PERCENT (in module *IoticAgent.Units*), 55
 pid (*IoticAgent.IOT.Point.Point* attribute), 34
 pid_check_convert() (*IoticAgent.Core.Validation.Validation* class method), 11
 Point (class in *IoticAgent.IOT.Point*), 31
 PointDataObject (class in *IoticAgent.IOT.Point*), 34
 PointDataObjectHandler (class in *IoticAgent.IOT.PointValueHelper*), 35
 PointMeta (class in *IoticAgent.IOT.PointMeta*), 35
 print_units() (*IoticAgent.Units.Units* method), 55
 private_name_for() (in module *IoticAgent.IOT.utils*), 50
 private_names_for() (in module *IoticAgent.IOT.utils*), 50
 proplist_check_convert() (*IoticAgent.Core.Validation.Validation* class method), 11
 PUBLIC (*IoticAgent.Core.Const.DescribeScope* attribute), 10
 PUBLIC (*IoticAgent.Core.Const.SearchScope* attribute), 10

R

RADIAN (in module *IoticAgent.Units*), 55
 reassign() (*IoticAgent.IOT.Thing.Thing* method), 48
 REDUCED (*IoticAgent.Core.Const.SearchType* attribute), 10
 RefreshException, 35
 register_callback_created() (*IoticAgent.IOT.Client.Client* method), 18
 register_callback_deleted() (*IoticAgent.IOT.Client.Client* method), 18
 register_callback_duplicate() (*IoticAgent.IOT.Client.Client* method), 19

register_callback_reassigned() (*IoticAgent.IOT.Client.Client method*), 19
 register_callback_renamed() (*IoticAgent.IOT.Client.Client method*), 20
 register_callback_subscribed() (*IoticAgent.IOT.Client.Client method*), 21
 register_callback_subscription() (*IoticAgent.IOT.Client.Client method*), 21
 register_catchall_controlreq() (*IoticAgent.IOT.Client.Client method*), 21
 register_catchall_feeddata() (*IoticAgent.IOT.Client.Client method*), 22
 RemoteControl (class in *IoticAgent.IOT.RemotePoint*), 35
 RemoteFeed (class in *IoticAgent.IOT.RemotePoint*), 37
 RemotePoint (class in *IoticAgent.IOT.RemotePoint*), 38
 rename() (*IoticAgent.IOT.Point.Point method*), 34
 rename() (*IoticAgent.IOT.Thing.Thing method*), 48
 Resource (class in *IoticAgent.IOT.Resource*), 38
 ResourceMeta (class in *IoticAgent.IOT.ResourceMeta*), 38
 RetryingThingRunner (class in *IoticAgent.ThingRunner*), 51
 ROENTGEN (in module *IoticAgent.Units*), 55
 run() (*IoticAgent.ThingRunner.ThingRunner method*), 52
 RunContext (class in *IoticAgent.ThingRunner*), 51

S

save() (*IoticAgent.IOT.Config.Config method*), 28
 save_config() (*IoticAgent.IOT.Client.Client method*), 22
 save_json() (*IoticAgent.Units.Units method*), 55
 search() (*IoticAgent.IOT.Client.Client method*), 22
 search_check_convert() (*IoticAgent.Core.Validation.Validation class method*), 11
 search_located() (*IoticAgent.IOT.Client.Client method*), 24
 search_location_check() (*IoticAgent.Core.Validation.Validation class method*), 11
 search_minimal() (*IoticAgent.IOT.Client.Client method*), 24
 search_property() (*IoticAgent.IOT.Client.Client method*), 24
 search_property_located() (*IoticAgent.IOT.Client.Client method*), 26
 search_property_minimal() (*IoticAgent.IOT.Client.Client method*), 26
 search_property_reduced() (*IoticAgent.IOT.Client.Client method*), 26
 search_reduced() (*IoticAgent.IOT.Client.Client method*), 26
 search_scope_check_convert() (*IoticAgent.Core.Validation.Validation class method*), 11
 search_type_check_convert() (*IoticAgent.Core.Validation.Validation class method*), 11
 SearchScope (class in *IoticAgent.Core.Const*), 10
 SearchType (class in *IoticAgent.Core.Const*), 10
 SECOND (in module *IoticAgent.Units*), 55
 send() (*IoticAgent.Core.AmqpLink.AmqpLink method*), 9
 set() (*IoticAgent.IOT.Config.Config method*), 28
 set() (*IoticAgent.IOT.ResourceMeta.ResourceMeta method*), 39
 set_description() (*IoticAgent.IOT.ResourceMeta.ResourceMeta method*), 40
 set_label() (*IoticAgent.IOT.ResourceMeta.ResourceMeta method*), 40
 set_location() (*IoticAgent.IOT.ThingMeta.ThingMeta method*), 49
 set_meta_rdf() (*IoticAgent.IOT.Point.Point method*), 34
 set_meta_rdf() (*IoticAgent.IOT.Thing.Thing method*), 48
 set_public() (*IoticAgent.IOT.Thing.Thing method*), 48
 set_recent_config() (*IoticAgent.IOT.Point.Feed method*), 30
 setup_logging() (*IoticAgent.IOT.Config.Config method*), 28
 share() (*IoticAgent.IOT.Point.Feed method*), 30
 share_async() (*IoticAgent.IOT.Point.Feed method*), 31
 shutdown_requested (*IoticAgent.ThingRunner.ThingRunner attribute*), 52
 simulate() (*IoticAgent.IOT.RemotePoint.RemoteFeed method*), 38
 simulate_feeddata() (*IoticAgent.IOT.Client.Client method*), 27
 SQUARE_CENTIMETER (in module *IoticAgent.Units*), 55
 SQUARE_METER (in module *IoticAgent.Units*), 55
 SQUARE_MILLIMETER (in module *IoticAgent.Units*), 55
 start() (*IoticAgent.Core.AmqpLink.AmqpLink method*), 9
 start() (*IoticAgent.IOT.Client.Client method*), 27
 stop() (*IoticAgent.Core.AmqpLink.AmqpLink method*),

9
stop() (*IoticAgent.IOT.Client.Client method*), 28
stop() (*IoticAgent.ThingRunner.ThingRunner method*), 52
str_to_foc() (*in module IoticAgent.IOT.utils*), 50
str_to_logging() (*in module IoticAgent.IOT.Config*), 28
STRING (*in module IoticAgent.Datatypes*), 14
subid (*IoticAgent.IOT.RemotePoint.RemotePoint attribute*), 38
sync_timeout (*IoticAgent.IOT.Client.Client attribute*), 28

T

tags_check_convert() (*IoticAgent.Core.Validation.Validation class method*), 11
tell() (*IoticAgent.IOT.RemotePoint.RemoteControl method*), 36
tell_async() (*IoticAgent.IOT.RemotePoint.RemoteControl method*), 37
Thing (*class in IoticAgent.IOT.Thing*), 40
ThingMeta (*class in IoticAgent.IOT.ThingMeta*), 49
ThingRunner (*class in IoticAgent.ThingRunner*), 51
TIME (*in module IoticAgent.Datatypes*), 14
to_dict() (*IoticAgent.IOT.Point.PointDataObject method*), 34
type_ (*IoticAgent.IOT.PointValueHelper.Value attribute*), 35

U

unattach() (*IoticAgent.IOT.Thing.Thing method*), 49
unfollow() (*IoticAgent.IOT.Thing.Thing method*), 49
unit (*IoticAgent.IOT.PointValueHelper.Value attribute*), 35
Units (*class in IoticAgent.Units*), 55
unset (*IoticAgent.IOT.PointValueHelper.Value attribute*), 35
unset() (*IoticAgent.IOT.Point.PointDataObject method*), 34
UNSIGNED_BYTE (*in module IoticAgent.Datatypes*), 14
update() (*IoticAgent.IOT.Config.Config method*), 28
update() (*IoticAgent.IOT.ResourceMeta.ResourceMeta method*), 40
URI (*in module IoticAgent.Datatypes*), 14
uri_ref_check_convert() (*IoticAgent.Core.Validation.Validation static method*), 11
uuid_to_hex() (*in module IoticAgent.IOT.utils*), 50

V

Validation (*class in IoticAgent.Core.Validation*), 10
Value (*class in IoticAgent.IOT.PointValueHelper*), 35

value (*IoticAgent.IOT.PointValueHelper.Value attribute*), 35
value_type_check_convert() (*IoticAgent.Core.Validation.Validation class method*), 12
value_unit_check_convert() (*IoticAgent.Core.Validation.Validation class method*), 12
values (*IoticAgent.IOT.Point.PointDataObject attribute*), 34
version_string_to_tuple() (*in module IoticAgent.IOT.utils*), 50
VOLT (*in module IoticAgent.Units*), 55

W

wait_for_shutdown() (*IoticAgent.ThingRunner.ThingRunner method*), 52
WATT (*in module IoticAgent.Units*), 55
WATT_HOUR (*in module IoticAgent.Units*), 55