

---

# **aioLXD**

***Release 0.0.1***

**Apr 21, 2019**



---

## Contents

---

<b>1</b>	<b>aiolxd.remote</b>	<b>1</b>
<b>2</b>	<b>aiolxd.uri</b>	<b>3</b>
<b>3</b>	<b>aiolxd.api.http</b>	<b>5</b>
<b>4</b>	<b>aiolXD - Asynchronous client library for the LXD API</b>	<b>7</b>
4.1	Indices and tables . . . . .	8



# CHAPTER 1

---

aiolxd.remote

---



## CHAPTER 2

---

aiolxd.uri

---





## CHAPTER 3

---

aiolxd.api.http

---



---

## aiolXD - Asynchronous client library for the LXD API

---

aiolXD is an `asyncio`-based client library for the [LXD REST API](#).

It provides an high level API to interact with resources on LXD servers, such as containers, images, networks, profiles and storage.

LXD servers are accessible through the `aiolxd.remote.Remote` class, which exposes server details and configuration, as well as access to resource collections.

Collections (such as *containers*, *images*, *profiles*, *networks*, ...) allow creating and fetching resources, which can be modified, updated or deleted.

For example:

```
from pprint import pprint

from aiolxd import lxc

# get all remotes defined in the client config
remotes = lxc.get_remotes()
async with remotes['local'] as remote:
    # fetch all images and print their details
    resp = await remote.images.read()
    for image in resp:
        resp = await image.read()
        pprint(resp.metadata)
        # image details have been read, now they're also cached (same
        # output as above)
        pprint(image.details())

    # fetch a single container by name
    container = await remote.containers.get('c')
    pprint(container.details())
    # rename it
    await container.rename('new-c')
```

(continues on next page)

(continued from previous page)

```
# change some details
await container.update({'description': 'foo'})
# and now delete it
await container.delete()
```

## 4.1 Indices and tables

- [genindex](#)
- [modindex](#)
- [search](#)