
{{cookiecutter.project_name}} **Documentation**

{{cookiecutter.author}}

Dec 29, 2019

Contents

1	Introduction	3
1.1	Installation	3
1.2	Client	3
1.3	TL;DR	3
1.4	Not Long Enough;Will Read	4
2	Changelog	5
2.1	1.0.0	5
3	Indices and tables	7

Contents:

> Simple server to provide an endpoint used by pact to generate a provider state

1.1 Installation

1. Install the package:

```
pip install pact_state_provider
```

1.2 Client

Entrypoint `pact-state-provider`:

```
Usage: pact-state-provider [OPTIONS]

Start the state provider server on the specified host and port.

Options:
  --base-module TEXT  Module containing the state providers.
  --host TEXT         Host for the endpoint.
  --port INTEGER      Port for the endpoint.
  --help              Show this message and exit.
```

1.3 TL;DR

Example:

```
pact-state-provider --base-module my_provider.states
# pact requests a state:
# GET http://127.0.0.1:1235 {'consumer': 'TestConsumer', 'state': 'user exists'}
-> function "my_provider.states.user_exists('TestConsumer')" is executed
```

1.4 Not Long Enough;Will Read

When verifying contracts with **Pact** a state provider is called for the particular consumer with the state requested. A consumer `TestConsumer` with a contract starting with `given('user exists')` requests the configured endpoint with the payload:

```
{
  'consumer': 'TestConsumer',
  'state': 'user exists'
}
```

In order to execute the specific provider functions without having to implement a dedicated endpoint on the provider or a dedicated application with provider specific code `pact-state-provider` provides a simple http server endpoint which calls an existing module function based on the request payload.

In the given example a function called `user_exists` (invalid characters are translated to underscores) will be executed on the specified base module (`--base-module` parameter).

This gives the user to maintain the state provider code in the actual provider codebase, but easily have an endpoint to use with `pact`.

CHAPTER 2

Changelog

2.1 1.0.0

- Initial version

CHAPTER 3

Indices and tables

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