
wharfrat Documentation

Release REL

Julian Phillips

Sep 30, 2018

Contents:

1	Introduction	1
2	Installation	3
3	Basic Usage	5
4	Configuration	7
5	Accessing Containers	9
5.1	wharfrat run	9
5.2	Exposing Commands	9
6	Examples	11
7	Indices and tables	13

CHAPTER 1

Introduction

wharfrat is intended to make it easy and convenient to use a development environment in a docker container. The benefits of this are:

- **Simple:** A new development environment is setup with a single simple command, meaning new team members are ready to go immediately.
- **Shared:** Everyone with access to the project (and docker) has access to the development environment.
- **Controlled:** Everyone gets a development environment created from the same image - no more “works for me” issues.
- **Versioned:** The configuration is version controlled, meaning you get the development environment that matches the code branch you are working on.

CHAPTER 2

Installation

...

CHAPTER 3

Basic Usage

...

CHAPTER 4

Configuration

...

Accessing Containers

Table of Contents

- *Accessing Containers*
 - *wharfrat run*
 - *Exposing Commands*

5.1 wharfrat run

5.2 Exposing Commands

It is possible to use the `wr-exec` command to expose commands from inside a container to the host. This is normally done by creating an executable config file with a `#!` line that invokes `wr-exec`. For example, if `./test` contains:

```
#!/usr/bin/env wr-exec

project = "/path/to/project/file"
command = ["command", "arg1"]
```

Then, running `./test arg2` will run the command `command arg1 arg2` in the container for the default crate defined in the wharfrat project file at `/path/to/project/file`.

CHAPTER 6

Examples

...

CHAPTER 7

Indices and tables

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