

Moonraker Final Documentation

Chance Murray, Dan Kindt, Gabe Coelho, Grant Perdue & Madison Robertson



Abstract

Working alongside Microsoft, our team has delivered a visualization for Cloud Native Application Bundles. Our objective was to create a tool to help users view the data associated with their Cloud Native Application bundles by April 10, 2020. Using the latest collaboration tools we developed a Command Line Interface that sets up a web server and displays a dashboard on a local host. We successfully met the minimum viable product and managed to iterate the front-end to exceed our sponsor's expectations.

Introduction

Moonraker is a visualization tool for Cloud Native Application Bundles that watches the claims files produced by Porter, Duffle and Docker-App, and then aggregates the data provided into digestible metrics in the form of a dashboard. The idea is to display the data on a IT Operations center screen so that teams can know the status of their bundles at a glance. Anyone setting it up only needs to install the tool, and run two commands to have the dashboard setup.

Body

The team built a CLI tool and a dashboard. The CLI includes an API that the dashboard gets data from. The job of the dashboard is to just display the data it gets from the API. For this product, it is assumed that a user (stakeholder) has configured their CNAB installers to store data in a remote (cloud) datastore as well as local data.

The API and frontend will both be distributed via bundles which will include the binary for the backend and a packaged version of the front end. The users will then configure their cloud storage information and then run the service. Any device on the network will then be able to consume the exposed API and view the hosted dashboard.

Major Components and Functions

1. Dashboard (front-end user interface)
2. Fully documented API (backend and data gathering)

3. Easy install CLI command

Stakeholders

- Anyone that has cloud-native application bundles installed
- DevOps and infrastructure teams
- Site reliability engineers
- System administrators
- Anyone deploying applications using cloud-native application bundles

Problems Moonraker Solves

- Gathers and provides information about CNAB installations
- Creates an easy-to-read dashboard
- Sets up data to automatically update packages (in the future)

Other Contributors

- Open source developers
- Microsoft Azure Open Source Dev Team
- CNAB open-source project members

Conclusion

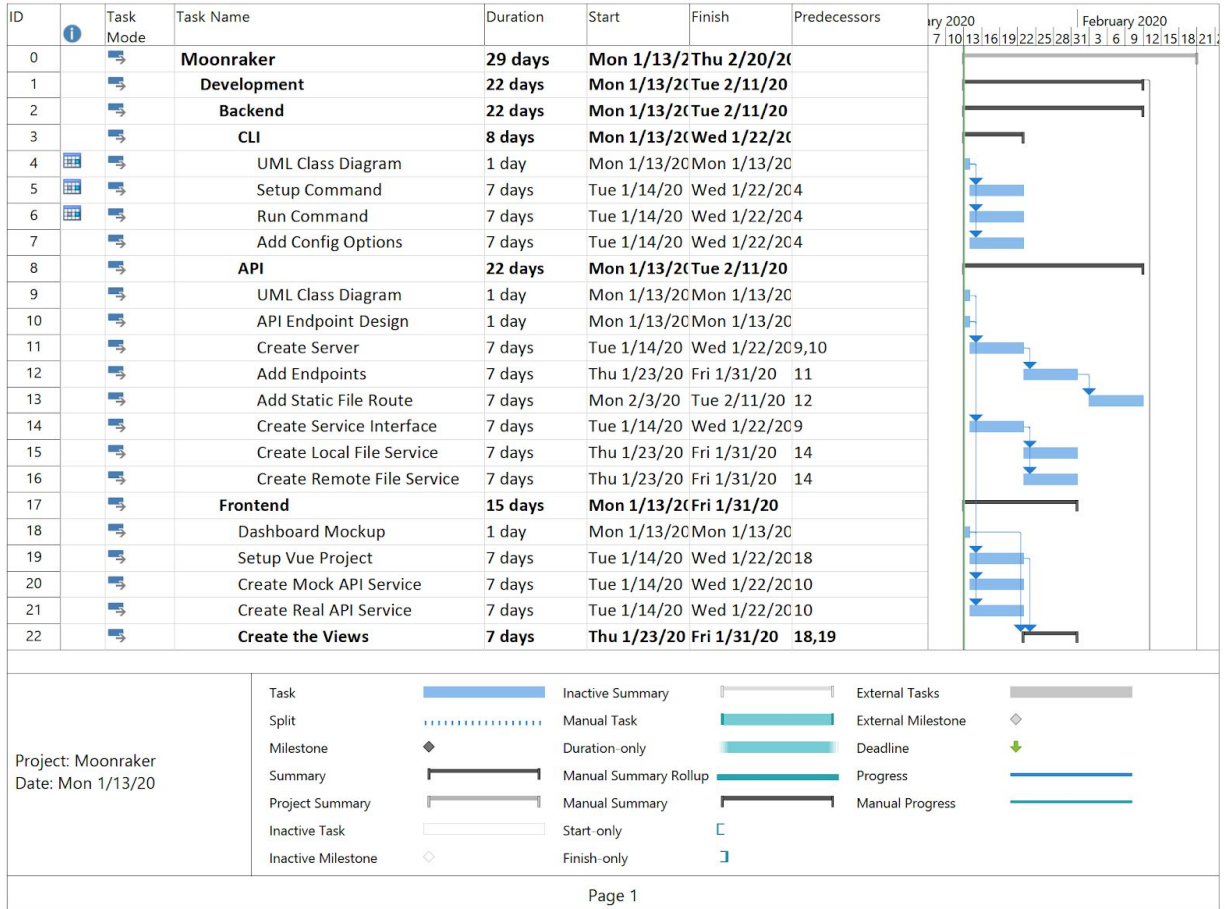
Moonraker successfully visualizes Cloud Native Application bundles. Moonraker is only limited by the data available from the CNAB installers. As our sponsors and the open-source community continue to develop CNAB and CNAB installers, Moonraker will be able to capture and display more relevant data, specific to the end users. In addition, future capstone groups might be able to help the open-source community by enhancing Porter, Duffle and docker-app, which would enable Moonraker to grow and become a better tool.

Bibliography

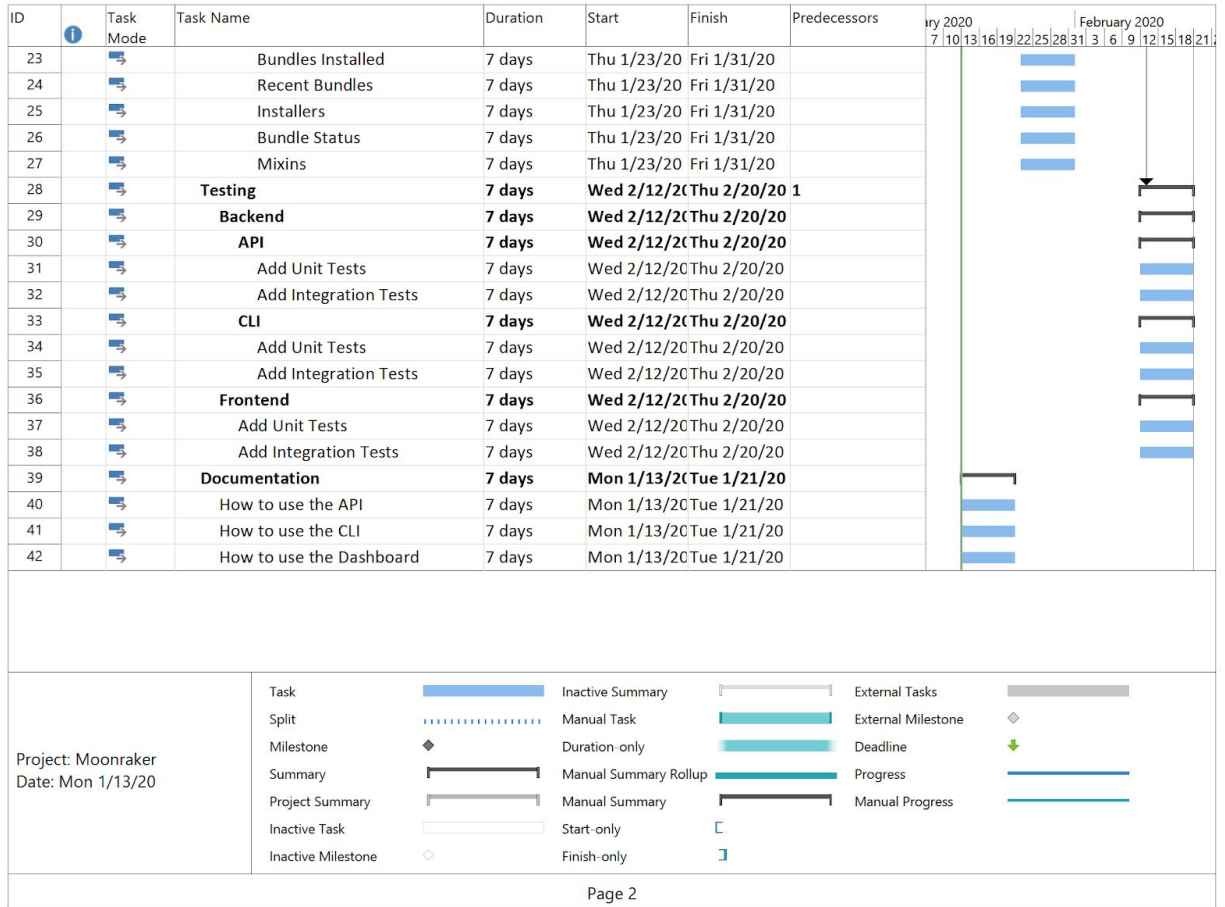
CNAB: a spec for packaging distributed apps. (n.d.). Retrieved from <https://cnab.io/>
Wallen, J. (2019, November 6). What Is CNAB and Why It Is Important for Cloud Native Computing. Retrieved from <https://thenewstack.io/what-is-cnab-and-why-it-is-important-for-cloud-native-computing/>

Appendix


I. Gantt Chart #1



II. Gantt Chart #2





III. Poster



Moonraker

Visualizing Cloud Native Application Bundles

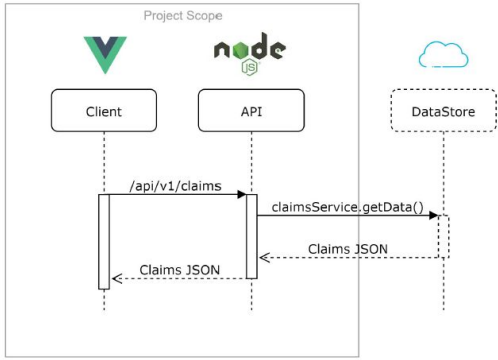


Problem

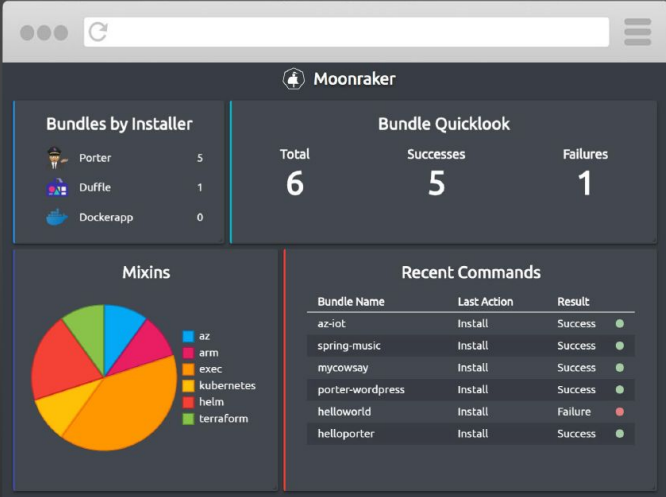
Cloud Native Application Bundle (CNAB) tools create separate files containing the key data for each bundle. These files are called "Claims." Viewing data from all of the Claims at once is currently impossible.

Solution

Moonraker is a CLI tool that gathers all the data from the Claims and serves it as an HTTP API. Moonraker also provides a dashboard to consume the API and visualize the key data from Claims.



```
graph TD
    subgraph Project_Scope [Project Scope]
        direction TB
        subgraph Client_API [ ]
            direction LR
            Client[Client]
            API[API]
        end
        DataStore[(DataStore)]
    end
    Client -- "/api/v1/claims" --> API
    API -- "claimsService.getData()" --> DataStore
    DataStore -- "Claims JSON" --> API
    API -- "Claims JSON" --> Client
```




Bundles by Installer

Installer	Count
Porter	5
Duffle	1
Dockerapp	0

Bundle Quicklook

Total	Successes	Failures
6	5	1

Mixins




Recent Commands

Bundle Name	Last Action	Result
az-iot	Install	Success
spring-music	Install	Success
mycowsay	Install	Success
porter-wordpress	Install	Success
helloworld	Install	Failure
helloporter	Install	Success

Sponsors Taylor Thomas & Carolyn VanSlyck

Coach David Carr

Team Madison Robertson, Chance Murray, Gabe Coelho, Grant Perdue & Dan Kindt



Capstone 2020

IV. More about CNAB Handout



More About CNAB

Definition

A Cloud Native Application Bundle (CNAB) is an open source, cloud-agnostic specification for packaging and running distributed applications. It facilitates the bundling, installing and managing of container-native apps and their dependent services. It describes a technology for bundling, installing, and managing distributed applications, that are by design, cloud agnostic.

Use case

Modern applications are made up of a wide range of components and services — they can be multiple cloud resources, managed services, SaaS offerings, containers, configuration formats, functions, and more. CNAB pulls these disparate components together, providing a common packaging format for multiservice applications. These bundles can be developed, managed and shared (across a registry like Docker Hub) as one immutable composite unit without forcing any specific environment/clouds. (From article linked below)

What Is CNAB?

<https://bit.ly/2DIN9fW>



CNAB Official Website

<https://cnab.io>



V. Final Video

<https://bit.ly/3ect8PS>