Ben Fickes

English 3315

Tom Akbari

Unit 1 Rough Draft

13 September 2019

An Analysis of Buildpacks

I will be analyzing the Cloud Native Buildpacks website, which I came across during my first co-op as a software engineer working on Pivotal’s buildpacks team. As a bit of background, buildpacks are used to compile projects (denoted “apps”) pushed to the cloud using cloudfoundry, an open source project. Buildpacks, being built on cloudfoundry, are also open source. This buildpacks initiative was undergone collaboratively by both Pivotal and Heroku. Though this project itself is open source, both Pivotal and Heroku have proprietary “value-added” components that work on top of buildpacks. So, in addition to the “altruistic” goal of improving open-source software, these two companies, who worked on the website, have personal stake in enticing developers to use buildpacks, as they may then consider purchasing further resources from these companies. With this background, I will examine the contents of the website and discern what it is trying to communicate.

Specifically, I wish to point out that, while there is some technical mention, the primary goal of this page is not to demonstrate the efficacy of the buildpack, rather than explain how it works. I will examine the “What are buildpacks?” section, since it seems to be the clearest indicator of what the website is seeking to tell its viewers. In particular, consider the second supporting point: “[Buildpacks e]nsure that apps meet security and compliance requirements without developer intervention.”1(par. 3) This advocates for two benefits of buildpacks: meeting security requirements and not requiring developer intervention. Neither of these are technically dense claims. While one might argue that this lack of technicality indicates that the intended audience is not developers, I would disagree. In fact, the very next section contrasts buildpacks with Dockerfiles,1(par. 4) a point that is meaningless without at least cursory awareness of what Dockerfiles are and how they work. I would argue that the lack of technical specificity of this point actually supports its main boon: abstraction. The purpose of a buildpack is to accomplish its goal (meeting compliance requirements) without any developer intervention.1(par. 3) While a developer could concern themselves with the particular workings of buildpacks (and, since the buildpack is open source and on Github, this is very easy to do), the point of the buildpack is to do its work effectively without any mental strain on the developer. Thus, simply promising to take care of an end goal without saying how is actually its desired purpose. Further supporting this is the specific wording chosen: “meet security and compliance regulations.”1(par. 3) Rather than emphasizing that the buildpacks will make an app secure, the site emphasizes that the buildpacks will make an app meet security regulations. Though the two statements are similar, the subtle difference lies in the promise of taking responsibility from the developer. If the line had promised to allow developers to make apps secure, it would have marketed buildpacks as a useful tool. Instead, it markets as something to automatically meet external requirements placed on the developer. In other words, rather than promising to fix a problem, it promises to let the developer ignore the problem. A developer, whose primary goal of software is most likely not security, is promised to be able to focus on their goal without needing to consider whether or not they meet their organization’s security requirements.

This webpage showcases a standard approach in sharing new software; many new software applications are built to solve incredibly specific problems, and the Cloud Native Buildpacks are no different. The site seeks to concisely state the specific problem these buildpacks solve, while trying to state the solution as simply and accessibly as possible, in order to have the broadest appeal possible for this product. I believe this page succeeds at its goal. While not neglecting to speak of the specific issue buildpacks solve, allowing for both “OS-level and application level dependency updates,”1(par. 4) it does so in simple terms that showcase the “what,” not the “how” of the software, making the product’s benefit enticing and clear to developers, its target audience.

**Reference**

1. Cloud Native Buildpack Documentation [Internet]. N.p, n.d. [accessed 11 September 2019]. https://buildpacks.io/