Ben Fickes
English 3315
Tom Akbari
Unit 1 Final Draft
20 September 2019

An Analysis of Buildpacks

Software has become more and more pervasive in the modern world; some of it is marketed to the general public, but even more of it seeks to fill relatively niche, context-specific needs. As consumers discover and use software, for both personal and professional reasons, it is beneficial to examine how software companies communicate what their software does and whom that software is for. 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, and how it advocates for its software's relevance and usefulness. As a bit of background, buildpacks are software used to compile developers' projects (denoted "apps") pushed to the cloud-based platform known as cloudfoundry. Buildpacks and cloudfoundry are both open source, meaning the code is freely available to the public, and anyone, with permission from moderators, can contribute to it. This Cloud Native Buildpack initiative was a 2018 collaborative effort by both Pivotal and Heroku, two companies that specialize in a variety of cloud-based software services. Though this project itself is open source, both Pivotal and Heroku have proprietary components they sell that work in synchronization with buildpacks. So, in addition to the altruistic goal of improving open-source software, these two companies, whose engineers are responsible for the content of the website, have personal stake in enticing developers to use buildpacks; these developers 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, while there is some technical mention, the primary goal of this page is to demonstrate the efficacy of buildpacks, rather than explain how they work, and to market them to developers. In particular, consider the second supporting point of the "What Are Buildpacks?" section: "[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; that is, they are silent on program design choices and how features are implemented. One might argue that this lack of technicality indicates that the intended audience is not software developers, but this is not the case. In fact, the very next section contrasts buildpacks with Dockerfiles,[1(par. 4)] a point that is meaningless without a software background to understand Dockerfiles (configurability files for apps that utilize a popular platform called Docker) at a conceptual level. 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 buildpacks are open source and on Github, this is very easy to do), the point of the buildpack is to do its work effectively without forcing any mental strain on the developer. Thus, simply promising to take care of an end goal without saying how is the desired purpose of Cloud Native Buildpacks. 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 tool to more easily enact specific strategies to prevent malicious actions against the app. Instead, it markets buildpacks as something to, without developer effort, meet external requirements placed on the developer. In other words, rather than promising to fix a problem, it promises to let the developer, whose primary goal of software is most likely not security, ignore the problem and focus on the end goal of their app. Through this, software developers, even with their programming knowledge, can understand how Cloud Native Buildpacks affect them without being required to dive into and understand an entire project's worth of code.

This Cloud Native Buildpacks webpage showcases an effective way to advertise new software to potential users. While my Northeastern undergraduate experience has taught me how to write good code and how to document it in a way that allows other people working on the same project to interact with it, it has not given me extensive preparation in how to explain projects to those not involved with working on them. This is understandable, as, most likely, upon graduation I will work on someone else's project and be responsible for its implementation, not its vision or how it is marketed to others. But, should I ever create or design my own software, I will need to understand how to communicate what it does clearly and appealingly. I came across the Cloud Native Buildpacks site at my first co-op, and my second co-op did not entail creating software designed for developers, so I don't have firsthand experience working with a comparable marketing of software to software developers. However, the Cloud Native Buildpacks site concisely states the very specific problem these buildpacks solve (allowing for both "OS-level and application level dependency updates"[1(par. 4)]), while trying to state the solution as simply and accessibly as possible, in order to have the broadest appeal possible for this product. If I ever create my own software that I wish to showcase to the world, I hope to communicate its effectiveness as clearly and accessibly as the Cloud Native Buildpacks site.

Reference

1. Cloud Native Buildpack Documentation [Internet]. Cloud Native Sandbox, 2018 [accessed 11 September 2019]. https://buildpacks.io/