**BEAKER**

Christian Klein

Joseph Thill

Jay Patel

Derek Yu

Zifeng Jiang

**Summary**

Beaker is a notebook style environment for visualizing, documenting, and analyzing large, complicated data sets. This project supports switching between programming languages, and if a language is unknown to the system, it can easily be supported within the architecture. Beaker follows a style of development helpful for analyzing data, which utilizes sets of code blocks known as cells. These cells can act independently or together. A developer can experiment in one cell, then write more code, in a different language, dependent on the previous results. This ability to switch languages allows Beaker to take advantage of each language's strengths and leads to ‘iterative exploration’, where scientists can identify common patterns or artifacts within the data. Their mission is to promote, manage and maintain open source software projects.

**Purpose**

Beaker used to be a single-language platform, but it now supports many languages: Python, Python3, R, Julia, JavaScript, SQL, Java, Clojure, HTML5, Lau/Torch, Node.js, C++, TeX, Ruby, Scala/Spark, Groovy, and kdb make up just a small subset of supported languages. The project is open-source with lead developers gatekeeping the code submissions.

The IDE itself is composed of multiple coding blocks allowing the user to analyze, visualize, and document data while utilizing a multitude of programming languages. Beaker’s autotranslate lets a developer declare specific variable in a cell in one language. Beaker allows notebooks to change their sections in hierarchical sections. The older version of Beaker had a sharing server that used Github to host the content. It now also works on all the operating systems including Mac, linux, windows and browsers including Chrome, Safari, Mozilla. The application itself capable of being both a cloud-based server and a standalone executable that can be downloaded and ran.

**Stakeholders**

Beaker’s largest stakeholders are the code developers. Since Beaker is open sourced, once a developer has access, it is easy to contribute and improve the project. There are several people who have committed on this project and are trying to improve the software. Scottdraves, mnavasiolava, and anglee are a few developers who have heavily contributed to the project. They are a part of Two Sigma Open Source, where they manage, promote and maintain open source software projects.

They have been making continuous improvement in the software so that people can use it more frequently and get a good practice of coding/reading and debugging other people’s code.

The end-users of the software should be programmers who master many languages since Beaker is a tool for switching the code language easily. They can use it as their best integrated development environment.

Other customers include the developers who are interested in this tool and participate into the development progress. If they do some meaningful improvement on Beaker, they can also be regarded as contributors. People are also interested in improving their code literacy from an existing codebase could pull the source code from Git and start their study.

The developers are mainly from Two Sigma Open Source, LLC. However, it is an open source

program so anyone can see the code from Git and fix bugs or make changes to the project after

signing the contract form.

Their developing is not for benefiting the company. They claim it on the website that Beaker is absolutely free and nobody is able to earn profit from it since the interpretation and the right are on their side.

There is no direct words saying that Beaker is from the research laboratory. Beaker is built upon other open source projects such as Angular, Bootstrap, CometD, Gradle, Guice, IPython, Jackson, Jetty, Jackson, Nginx, Rserve, etc.

**Project Activity**

After looking at the project on Git, there are few files that are very old and some of them are newly updated. There are in total of 44 contributors in this project, most of them joined this project from very starting but some of them started in the middle of the project. There also have been a lots of commits from different developers, it looks like some of them are assigned different tasks and once they made some progress they are committing it to Git. There are also several branches in this project as well. After looking at the branches it looks like there are a lot of developers that branched the code and most of them are very recently updated.

The project has been updated recently since there are some activities that were done in the past hour or past day. There are very few issues that are resolved but after reading the comments it looks like there are a lot of methods in the project that still need a lot of improvement in order to get it to function. There are also a lot of comments on the projects, after reading them it looks like they are mentioning the errors that they are facing, letting others know what the issues are in order for others to resolve if they know the error, and letting others know about the difficulty they have been facing. As far as forks and likes are concerned there are 138 forks done in total. That means that this software was copied into a lot of repositories, and this will allow someone to freely experiment with changes without affecting the original project. There are in total of 993 stars for this project, which means a lot of people found this project very interesting. The most interesting part of this project is that there are 109 watchers in this project, which means a lot of people are interested on what is going to be updated and want a notification when commits, tagging, repository creation, etc is being changed.

**Contributors**

Beaker has a fair amount of developers contributing towards the project, as of now there are a total of 44 contributors on the GitHub page. The graph in GitHub indicates that the developers’ contributions have been increasing steadily through 2016, but have dropped slightly since then. After looking at history for all the commits it looks like, there are several developers in this project who have done most of the work but Scott Draves who is currently working at Two Sigma and is a founder of ElectricSheep.org has the most commits for this project. Mr. Draves has done a great job on this project since he has been continuously working on it since 2014. After Mr. Draves, another developer named Marina Navasiolava has the second largest number of commits for this project whereas this developer started working in Mid July 2015 and has done a fair amount of commits for this project since then. This seems to comes from a developer-contributor hierarchy, where any developer can contribute. But, their work is first submitted to the core developers for review before any actual code-push is made for the project.

**Development Process**

Developing and extending Beaker is highly encouraged by the Beaker team. As a developer creates their potential Beaker improvements, they are able to test their work in two major ways: end-to-end tests and performance tests. These tests are run with Protractor, a trusted AngularJS testing framework used within many open-source projects. Considering these testing options and the test-driven development process used by the Beaker team, Beaker’s 8000 commits can be trusted. Code improvements can be submitted to the Beaker team and to be reviewed and added accordingly.

Beaker’s 60 releases tend to follow a monthly schedule. Each release includes new and improved features as well as resolved issues found by users. All of the release notes can be found on Beaker’s GitHub page.

The project’s build process is simply a matter of following a step-by-step document specific to your desired operating system. The process involves a fair amount of package installations, but many of them are standard development tools commonly used throughout other projects.

**Com S 362 — Object-Oriented Analysis and Design**

**Certification of Individual Contribution and Understanding Form**

Project Name: Project Description Team Facilitator: Jay Patel

Homework Set: 1 Date: 09/03/2016

Directions: Enter each team member’s name, and a “work rating” (1 - 5 with 5 as high and 1 as low score) that corresponds to the relative share of work done by the team member. Comments to explain work ratings other than 3 must also be provided. The total of the ratings must add up to 3 times the number of team members. Each team member must sign at the end of the form showing your agreement to your team member’s contribution.

*The work rating will have a direct and significant effect on each student’s project and ultimately course grade. In fairness to all students, each team member must give this rating their attention.*

Also, each team member should individually rate the percentage that they understand of the solution. (This does not affect the grade, but is used as a means of communication.)

|  |  |  |  |
| --- | --- | --- | --- |
| Printed team member’s name | Work Rating  (1 - 5) | Comments / Explanations | Understanding Self-Rating  (0 to 100%) |
| Jay Patel | 3 |  | 90% |
| Christian Klein | 3 |  | 90% |
| Joseph Thill | 3 |  | 90% |
| Derek Yu | 3 |  | 90% |
| Zifeng Jiang | 3 |  | 90% |
| Total: | 15 |  | |

Note that you must make Total = 3 \* (Number of team members)

|  |
| --- |
| I agree to the above ratings and understand our team’s solution. (Each team member should sign their name below.) |
| Christian Klein |
| Jay Patel |
| Derek Yu |
| Zifeng Jiang |
| Joseph Thill |