Smart-bot enabled GitInsights for Business Intelligence

Divya Nambiar  
*SDCG*  
*Hewlett Packard Enterprise*Bangalore, India  
dnambiar@hpe.com

Upadhya, Avinash  
*SDCG*  
*Hewlett Packard Enterprise*Bangalore, India  
avinash.upadhya@hpe.com

Vikrant Dhimate  
*SDCG  
Hewlett Packard Enterprise*Bangalore, India  
vikrant.mah.dhimate@hpe.com

Rupesh Shantamurty  
*SDCG   
Hewlett Packard Enterprise*Bangalore, India  
rupeshs@hpe.com

*Abstract*— with the current adoption of CICD (Continuous Integration & Continuous Deployment) there is huge requirement to correlate the software development stages to product management. A lot of times the link between the two never gets established leading to delays in resolving customer issues and duplication of efforts.

Also, in the entire development process for a software, code changes and it changes a lot. A lot of code gets written and modified on daily basis. Once the product matures engineers realize that some part of the software should have been architected and designed differently. With the ever changing structure of development teams it is difficult to retain the context of most of the changes done along the way. So the challenge development teams face is in identifying the components of the software that should be undertaken for redesign or change or how to correlate changes causing issues which are managed as a part of defect management tools. To address this challenge we are proposing a framework in this paper which analyzes the trends in development lifecycle and presents the components that should be taken up for change. Also the solution provides smart bot solution to integrate development with workflow and product management tools like Rally, Jira, QuIX, etc. We call this framework “Smart-bot enabled GitInsights” since it is based on Git version control system and smart-bot to integrate the data obtained with product management tools. This framework will also assist software architects and product managers to make better decisions towards implementation of new features as well. Better decisions save time and expedite the product development lifecycle.

Keywords— Git, Analytics, D3, Machine Learning, Business Intelligence, bot, NLP, AI, Analytics

# Introduction

We at HPE write good software and that is the reason our software lasts for many years and sometimes lives for decades together. Such matured software undergoes multiple rounds of re-architecting and redesign. It becomes important to decide on which components need to be chosen for the redesign and re-architecting. The challenge is in identifying the components that are the trouble makes and that need change. Along with data/justification for choosing the component. Another challenge is how to use this data to relate and retrieve

Useful inferences to make optimized usage of resources, increased productivity and customer satisfaction

# Prepare Your Paper Before Styling

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting. Please note sections A-D below for more information on proofreading, spelling and grammar.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

# Our solution

## Trend Analysis on git repository

The solution provides Smart-bot enabled dashboard to perform trend analytics on the captured data, integrates and shares information across various product management and support tools. and presents it in a manner that is easily consumed by the end user.

## Smart-bot for Business Intelligence

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

## Abbreviations and Acronyms

Flask- Micro service

BI – Business Intelligence

AI- Artificial Intelligence

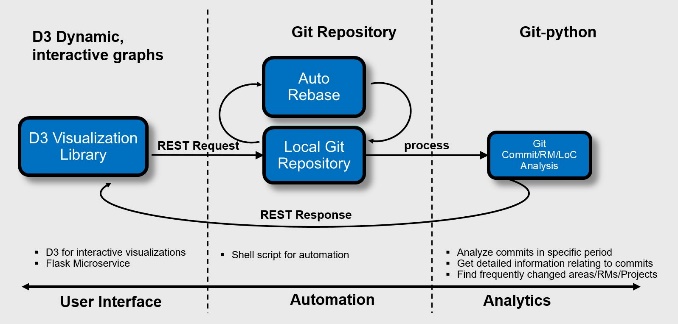
NLU – Natural Language Understanding

NLP - Natural Language Processing

NLG – Natural Language Generation

An interactive dashboard can be generated for identifying various trends like:

Below figure 1 show the basic design of our fremwork. Visualisation is done using D3 and the backend uses a combination of scripts and gitpython library.



* Most modified modules represented as a heat map of the modules modified.
* Most frequently changed modules
* Changes to a module requires changes to many other modules 🡪 representing loose / tight coupling between modules

The current advancement in the chat-bot frameworks and their integration with AI frameworks help us to solve the problem stated above. The AI frameworks are used to mine relevant data from various sources and then correlate them. The chat-bots help to interact with humans and present the mined data in a very simple and systematic way. Our solution lets users find and request services through a conversational and personalized interface. Prototyped chat-bot could effectively answer product lifecycle management related queries with an added advantage that it also provides filtered reports like defect summary by test failures, defect creation etc. So, the engineer gets the relevant information quickly without having to go through a series of web pages.



* A graph within a graph is an “inset”, not an “insert”. The word alternatively is preferred to the word “alternately” (unless you really mean something that alternates).
* Do not use the word “essentially” to mean “approximately” or “effectively”.
* In your paper title, if the words “that uses” can accurately replace the word “using”, capitalize the “u”; if not, keep using lower-cased.
* Be aware of the different meanings of the homophones “affect” and “effect”, “complement” and “compliment”, “discreet” and “discrete”, “principal” and “principle”.
* Do not confuse “imply” and “infer”.
* The prefix “non” is not a word; it should be joined to the word it modifies, usually without a hyphen.
* There is no period after the “et” in the Latin abbreviation “et al.”.
* The abbreviation “i.e.” means “that is”, and the abbreviation “e.g.” means “for example”.

An excellent style manual for science writers is [7].

# Using the Template

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

## Authors and Affiliations

**The template is designed for, but not limited to, six authors.** A minimum of one author is required for all conference articles. Author names should be listed starting from left to right and then moving down to the next line. This is the author sequence that will be used in future citations and by indexing services. Names should not be listed in columns nor group by affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization).

### For papers with more than six authors: Add author names horizontally, moving to a third row if needed for more than 8 authors.

### For papers with less than six authors: To change the default, adjust the template as follows.

#### Selection: Highlight all author and affiliation lines.

#### Change number of columns: Select the Columns icon from the MS Word Standard toolbar and then select the correct number of columns from the selection palette.

#### Deletion: Delete the author and affiliation lines for the extra authors.

## Identify the Headings

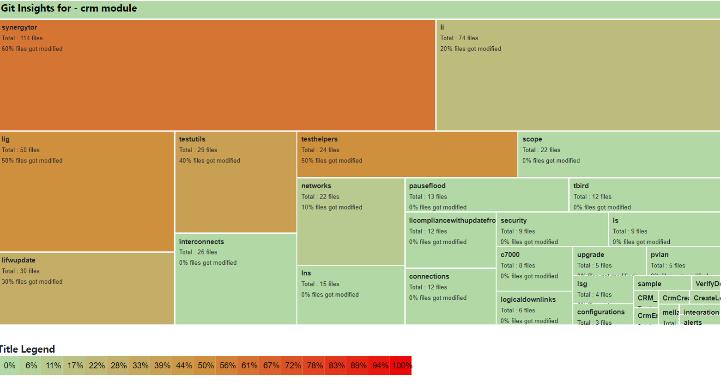
Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is “Heading 5”. Use “figure caption” for your Figure captions, and “table head” for your table title. Run-in heads, such as “Abstract”, will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named “Heading 1”, “Heading 2”, “Heading 3”, and “Heading 4” are prescribed.

## Evidence the solution works

#### We were able to quickly plug together the various modules as explained in the solution above and present a heat map of the changed module within two given dates. The POC is done in python using Flask Microservice, NLTK library and Rally APIs. The generic nature of chat-bot makes it easily integrate with other product management tools using REST API interfaces. The figure 2 below is a screenshot of the implemented proof of concept :



1. Implementaion of GitInsights showing heatmap of code changed between two dates



Fig. 2. Example of correlating existing features or enhancement requests with failure signature

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

1. GitPython - <https://github.com/gitpython-developers/GitPython> For backend implimentation
2. GitStats - <http://gitstats.sourceforge.net/>
3. Software development automated analytics - <https://patents.google.com/patent/US8745572B2/>
4. Paper from International Journal: <http://ijarcet.org/wp-content/uploads/IJARCET-VOL-4-ISSUE-10-3783-3785.pdf>
5. NLTK Source : <https://www.nltk.org/book>
6. <https://pythonprogramming.net/tokenizing-words-sentences-nltk-tutorial/>
7. <https://medium.com/analytics-vidhya/building-a-simple-chatbot-in-python-using-nltk-7c8c8215ac6e>
8. D3 Visualizations <https://d3js.org/> For frontend user interface