ACPAR (Analyzing Changes and finding Reasons for changes in refactoring PAtteRn )

**USER GUIDE:**

Pre-Requisites: Downloaded and running

1. JDK 1.4 or above
2. Eclipse version Juno or above.
3. Git To extract commit logs

For Pre Processing Data:

1. RefFinder Plug-in installation.

Input 1: Output from RefFinder plug-in

1. Commit log from Github or other version controlled system

Input2: Output from GitBash commit Log

1. Input 3: Official changes or release file.

Data Processing for reason statements and refactoring connection " \*.CSV" file:

1. Jar files for dependencies.
2. Run the jar file for the tool provided it can take 7-8 minutes to collect reason statement.
3. A GUI is displayed with 2 tabs one developer other user
4. Link for visualizations

**Pre-Requisites: Downloaded and running**

**JDK 1.4 or above**

get Java Development Kit and JRE from http://www.oracle.com/technetwork/java/javase/downloads/index.html

**Eclipse version Juno or above.**

Download Eclipse from .......https://eclipse.org/mars/

**Git To extract commit logs:**

https://git-scm.com/downloads

For windows: https://git-scm.com/download/win

installs GITBash and Git GUI within windows

**For Pre Processing Data:**

**RefFinder Plug-in installation.**

https://sites.google.com/site/reffindertool/[RefFinder Eclipse plug-in](https://sites.google.com/site/reffindertool/)

**Input 1: Output from RefFinder plug-in**

Step by step guide:

* + 1. Save base and current version in eclipse as new project.
    2. Make two new java projects in eclipse.
    3. Import saved base version and current version in these projects. Use General-->

existing file system--> saved versions.

* + 1. Open Refactoring view from Windows-->Show View-->Others-->

LSDiff-->Refactorings

* + 1. In Refactorings view click on folder icon and provide base and current versions.
    2. RefFinder will take 5-10 minutes depending on the size of project to generate a

rulebased file

**Commit log from Github or other version controlled system**

Clone Git repository for current version or version saved as current version for RefFinder.

**Input2: Output from GitBash commit Log**

*If using Git for windows:* After cloning rightClick on the folder and choose GitBASH

*In GitBASH type command :*

git log -p --pretty=format:":: commitID: %H - %s - %ad ::" >

Commit.txt

where git log : is command to get commit log for specific

directory we have cloned.

-p : is for paragraph view which shows coding also

for git commits

pretty format: is to show oneline format in customized string

%H: for whole commitID string,

%s:is for reason statement,

%ad: is for date of authoring the commit.

String after ">" stands for filename in which this log is saved.

**Input 3: Official changes or release file.**

Find the file "CHANGES.txt (for Cassandra versions)" in the Git repository

**Data Processing for reason statements and refactoring connection " \*.CSV" file:**

**Jar files for dependencies.**

We have added these jar files to project for dependencies:

..\poi-src-3.13-20150929.zip

..\poi-bin-3.13-20150929\poi-3.13\poi-3.13-20150929.jar

..\poi-bin-3.13-20150929\poi-3.13\poi-examples-3.13-20150929.jar

..\poi-bin-3.13-20150929\poi-3.13\poi-excelant-3.13-20150929.jar

..\poi-bin-3.13-20150929\poi-3.13\poi-ooxml-3.13-20150929.jar

..\poi-bin-3.13-20150929\poi-3.13\poi-ooxml-schemas-3.13-20150929.jar

..\poi-bin-3.13-20150929\poi-3.13\poi-scratchpad-3.13-20150929.jar

..\poi-bin-3.13-20150929\poi-3.13\ooxml-lib\xmlbeans-2.6.0.jar

..\org.eclipse.jgit-4.2.0.201601211800-r.jar

..\slf4j-1.6.0\slf4j-api-1.6.0.jar

..\slf4j-1.6.0\slf4j-ext-1.6.0.jar

..\slf4j-1.6.0\slf4j-simple-1.6.0.jar

..\jsch-0.1.50.jar

**Run the jar file for the tool provided it can take 7-8 minutes to collect reason statement.**

**A GUI is displayed with 2 tabs one developer other user**

**Link for visualizations :** http://renanpeixotox.github.io/visualization-tasks/pragya\_task/

(open in Chrome or fire-fox or mobile applications)