

## **What does BugBuilder do?**

BugBuilder is a tool to extract complete and concise bug-fixing patches from human-written patches in version control systems.

## **How to obtain BugBuilder?**

The replication package, including BugBuilder itself, can be downloaded from <https://github.com/jiangyanjie/BugBuilder> (This repository is permanently stored and publicly available).

## **What does the replication package contain?**

There are several folders within the replication package:

***/Implementation:*** The source code of BugBuilder (Java).

***/EvaluationData:*** The patches generated by BugBuilder in the evaluation as well as their explanations.

***/EvaluationData/matchedPatches:*** Patches generated by BugBuilder that are identical to those in Defects4J

***/EvaluationData/mismatchedAndUnconcisePatches:*** Unconcise patches generated by BugBuilder

***/EvaluationData/mismatchedButConcisePatches:*** Concise patches generated by BugBuilder that are different from those in Defects4J, and the justification/explanation for the conciseness (presented as README files accompanying the patches).

## **Dependency (tools/libraries required by BugBuilder):**

It depend on Defects4J to collect buggy source code and the manually constructed patches.

Detailed information of Defects4J can be found at

<https://github.com/rjust/defects4j#steps-to-set-up-defects4j>

Before running BugBuilder, please install and initialize Defects4J as follows:

- 1) Clone Defects4J:  
-- git clone https://github.com/rjust/defects4j
- 2) Initialize Defects4J:  
-- cd defects4j  
-- cpanm --installdeps .  
-- ./init.sh
- 3) Add Defects4J's executables to your PATH:  
--export PATH=\$PATH:"path2defects4j"/framework/bin
- 4) Check installation:  
--defects4j info -p Lang

## Step-by-step construction for running BugBuilder:

Illustrating with an example (Bug 26 of project **Lang**)

**Note:** we only test on Mac OS

1) Install and initialize Defects4J.

2) Preparation

■ Initialization of parameters in SpecialVersion.sh (Lines 22-25):

```
22  repoName="Lang" # Name of Java project containing the given bug
23  # Path to Defects4J,i.e.,where it has been installed
24  defects4jPath="/users/yanjiejiang/defects4j/framework/bin/defects4j"
25  BugBuilderPath="/Users/yanjiejiang/icse" # Path to BugBuilder, i.e., where it's source code has been placed
```

■ Initialization of parameters in Setting.java

```
1  public class Setting {
2      //Path to Defects4J,i.e.,where it has been installed
3      public static String defects4jPath="/users/yanjiejiang/defects4j/framework/bin";
4  }
```

3) Checkout source code to the specified version

■ cd BugBuilder/icse/src

■ ./SpecialVersion.sh

4) Run getDiffCommit.java

■ Javac getDiffCommit.java

■ Java getDiffCommit

5) Check the output:

If succeeded, the generated patch is stored in the section marked "succeed:".