| | | | | |
|---|---|---|---|---|
| | https://github.com/jagrosh/MusicBot | | | **include in report** |
| | | SLOC | Documentation Link | type how we performed the manual analysis |
| Version #1 | https://github.com/jagrosh/MusicBot/releases/tag/0.1.3 | | | show raw data of how we found it |
| Version #2 | https://github.com/jagrosh/MusicBot/releases/tag/0.2.0 | | | precision / recall values |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| **Conceptual Changes** | | | | |
| | **Change Log** | | | |
| **What's fixed?** | - Dependencies are now up to date<br>- Some edge-cases with blocking should be fixed | | | |
| **What's added?** | - Overhauled internals in preparations for expanding features easier in the feature<br>- Automatic DM to the bot owner when the new version is available (can disable in cofig)<br>- New flag to start in no-gui mode<br>- New config format, old configs will likely not work (Configuration is now in the HOCON format.) | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| **Changes** | Explanation | Change Type | Dependency Changes using CDA | TF-IDF |
| ownerCommand | This is a new abstract class that was added in to replace Command. The reason for this abstract class being added was to have a new category 'owner' for the owner of the server. The OwnerCommand class extends Command meaning it can do everything that Command can and adds another category attached to it. This adds an extra layer of hierachy to our software design. | | No dependency graphs extracted | Cannot extract the output csv files using TF-IDF because it is a new file. New files do not work with TF-IDF since we have no baseline version to compare the revised version with (it would be a blank output compared to the second version's output which is useless). |
| Listener | Listener is a class which was created specifically for the bot to listen in to the events that occur in the server. By implementing the class Listener, they removed the ListenerAdapter extensions from bot and the functions associated with it. This removed a lot of functionality from bot and moved it to a new, separate class (Listener) which manages the bot. | | JMusicBot.java uses Listener | The listenerAdapter was removed from the output file. This shows there is a high chance that the JMusicBot is no longer dependent on the listenerAdapter class. This aligns with the manual analysis |
| NowplayingHandler | This class was newly added to handle functions that existed in different classes. For example, we had updateTopic function which existed inside the bot originally and we moved it to the NowplayingHandler. There were also a small bit of logic change within the code. Fianlly, there were also a couple of new functions added in to handle different events that may occur. | | *QueueCmd dep AudioHandler<br>- NowPlayingHandler class does not exist in v1 (QueueCmd ->AudioHandler)<br>- AutoplaylistCmd class does not exist in v2 dep graph anymore.<br>- Listener abstract class does not exist in v1<br>- Classes PlaylistCmd$ListCmd, PlaylistCmd$DefaultlistCmd, PlaylistCmd$MakelistCmd do not exist in v2 | new file (see first example) |
| DJCommand | DJCommand replaced MusicCommand and it extends MusicCommand. This means that the DJCommand can do everything that MusicCommand can. By implementing the DJCommand, we removed the category from inside the bot and moved it to DJCommand for modularity. This means that we added an extra layer of hierachy to our software. | Hierarchical Change | *PauseCmd dep MusicCommand<br>- V2 now depends on DJCommand class<br>- ForceskipCmd class does not exist in v2<br>- VolumeCmd class does not exist in v2<br>- SkiptoCmd class does not exist in v2<br>- NowplayingCmd class does not exist in v2 | new file (see first example) |
| PlayerManager | PlayerManager replaced DefaultAudioPlayerManagers and it extends this instead. By having the PlayerManager, we can set up different handlers inside this class which we couldn't previously. This an extra layer of hierachy to our software. | | -Valerie's comment on how the dep graph is too big for manual comparison, and we will continue our analysis | new file (see first example) |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

| Part 3 analysis | what's changed | | | issues with using this tool |
|---|---|---|---|---|
| JMusicBot | The listenerAdapter was removed from the output file. This shows there is a high chance that the JMusicBot is no longer dependent on the listenerAdapter class. | | | 1, Runs the output on the entire file and all of it's content. This caused the output files to have lots of unnecessary words and symbols (such as =, and, or, etc.) which isn't the type of changes we are looking for. |
| | There were new additions to the file such as (create, deprecated). After taking a look at the Java documentations, it looks like there is no create / deprecated keyword meaning this may be detected from the comments. | | | 2. When the changes in the file is large, it becomes very difficult to compare. This is because the list of words used and the frequency of the words change drastically. As such, when mixed with comments, it is very difficult to keep track of what was actually changed. |
| | There were a lot of updates (addition and removal) of java packages used. Examples are: omjagroshjmusicbotcommandsgeneral, comjagroshjmusicbotcommandsmusic, comjagroshjmusicbotcommandsowner, comjagroshjmusicbotentitiesprompt | | | 3. We are not always sure of what some of the words mean. This is because we have just the word and the frequency of the word. If we have the name of a variable renamed, we can't see that this was just a simple rename and from the output, it looks like a portion of the code was newly created or was removed when in reality, it may just have been reformatted. This makes it hard to analyze what the type of changes actually were. |
| | listenerbot was removed in version 2 of JMusicBot. Initially we predicted that this was either a rename of an instance or a conceptual change (based on the name of the object). After taking a look at the code, this was actually 'new Listener(bot)' which was added in to handle how the bot listens in to events so this was a conceptual change. | | | 4. We have to compare each file separately and find the files with 'significant' changes and choose the files we want to run this algorithm on explicitly. |
| Bot | queuetrackcommandevent was removed in the second version. Since this is an event (judging by the name), there is a high chance that this was removed to be implemented in a different event handling class | | | 5. The frequency value of the words are difficult to compare. If total number of words decrease in the file but the number of certain words remain the same, then the frequency of the word increases which is an inaccurate representation. |
| | The onguildevents were removed in the second version. We suspect this is the case because the onguildevents was moved to a different class which handles the events. After performing manual analysis, we found out this was indeed the case. | | | 6. Trying to guess what changed is highly dependent on the naming of the variables. This is mainly due to the fact that we don't have much information other than the variable name and the frequency at which it appeared. |
| | A lot of setting variables were removed in the second version. We suspect this is the case because the settings are now handled in a different file instead of inside the bot (such as the botConfig). | | Dependency Changes using CDA | |
| | We found that there was an 'extends' removed in the second version. Because of this we suspect they removed the extend from the class declaration. However, from just the tf-idf information, we cannot figure out which dependency was lost. After performing manual analysis, we found the Bot class no longer extends ListenerAdapter | | No dependency graphs extracted | |
| | | | JMusicBot.java uses Listener | |
| | | | - NowPlayingHandler class does not exist in v1 (QueueCmd ->AudioHandler)<br>- AutoplaylistCmd class does not exist in v2 dep graph anymore.<br>- Listener abstract class does not exist in v1<br>- Classes PlaylistCmd$ListCmd, PlaylistCmd$DefaultlistCmd, PlaylistCmd$MakelistCmd do not exist in v2 | |
| | | | - V2 now depends on DJCommand class<br>- ForceskipCmd class does not exist in v2<br>- VolumeCmd class does not exist in v2<br>- SkiptoCmd class does not exist in v2<br>- NowplayingCmd class does not exist in v2 | |
| | | | -Valerie's comment on how the dep graph is too big for manual comparison, and we will continue our analysis | |

| | | | | |
|---|---|---|---|---|
| Github Project | https://github.com/processing/processing4 | | | |
| | | SLOC | Documentation Link | |
| Version #1 | https://github.com/processing/processing4/releases/tag/processing-1286-4.0.1 | | | |
| Version #2 | https://github.com/processing/processing4/releases/tag/processing-1287-4.0.2 | | | |
| | | | | |
| | | | | |
| **Conceptual Changes** | | | | |
| | **Change Log** | | | |
| **What's fixed?** | - OpenGL apps now work with macOS Ventura, #544.<br>- Library version number parsing wasn't ignoring comments properly, so libraries that followed the template closely were showing The version number for "…" is not a number. #586, #553.<br>- The Updates tab of the Contribution Manager was throwing error messages about this. progressBar. #567.<br>- The call to fullScreen() with pixelDensity(2) was broken with the default renderer. On startup, the sketch would report Display -1 does not exist, returning 1 for displayDensity(-1). #487.<br>- After creating 26 sketches, the "take a break" message still showed up after restarting Processing. #582<br>- With multiple users sharing a machine, Processing would not work properly for the second user due to a problem with the temporary directory. #549<br>- Added explanation for how to fix Processing when a user disallows access to the Documents folder on macOS. #581 | | | |
| **What's added?** | - Moved from SDK 11 to 17 | | | |
| | | | | |
| Classes/Files Changed | Explanation | Change Type | Dependency Changes using CDA | TF-IDF Analysis |
| Util.java | Library version number parsing wasn't ignoring comments properly, so libraries that followed the template closely were showing The version number for "…" is not a number. The readSettings function now takes a boolean parameter allowHex which handles the error thrown "The version number for "XXX" is not a number." An if statement checks if the string contains '#' and then trims the string accordingly, which in turn helps keep the version as numeric and Processing 4 would not reject the third party libraries. The following method calls to readSettings have their parameters adjusted as well. (fixes #586, #553) | If else statements added, 1 more parameter added to the method (semantic change) | No dependencies were found in this case. | AllowHex was added in the new version of the code, so it aligns our manual analysis |
| StatusDetail.java | The Updates tab of the Contribution Manager was throwing error message 'this.progressBar is null' which caused users unable to update their libraries. The update function now checks whether the progress bar exists or not through if-else statements. If the progress bar does not exist prior to the update function being called, the fucntion will now create progress bar to avoid null error completely. (fix #567) | If else statements added (semantic change) | No dependency graphs were extracted as v2 depends on JProgressBar() which is a library call. | The extracted words do not give enough information to find the changes to the if statement. "contribprogressprogressbar" exists in both files |
| PSurfaceJOGL.java | When using the P3D and P2D renderers, the window is not plotted on MacOS Ventura. With default size(), the window shows up correctly. The expected behaviour is the window should show up when using P3D and P2D. This was fixed by doing a very small change where we swapped the order of the method calls putting initDisplay() after initGL(). (fix #544) | Swapping method call orders. (semantic change) | No dependencies were found in this case. | The output of the TF - IDF tool does not provide enough clues to verify the changes we found manually. Both outputs are nearly identical except the new version has added the word "order". Which is not enough information to find any significant change without looking at the source code |
| ShimAWT.java | pixelDensity() is not working in Windows applications. The call to fullScreen() with pixelDensity(2) was broken with the default renderer. On startup, the sketch would report Display -1 does not exist, returning 1 for displayDensity(-1). For that, the method displayDensityImpl was initially taking cases when the parameter display was > 0 and <= displayDevices.length, but now it is handling more branches at display == -1, display ==SPAN, and display == 1 for all display devices available from an existing global devices array. (fix #487) | If else statements added to cover all possible cases. (semantic change) | No dependencies were found in this case. | After comparing the results provided by the tool, the "displaydensityimpli" word was added into the new version of the ShimAWT.java file |
| SketchName.java/ContributionManager.java | In the previous version, macOS users might not be able to restart Processing due to read/write permission for the modes model. Even though this bug was caused by mishandling from the user, this version offers more in-depth explaination to fix Processing when a user disallows access to the Documents folder for macOS users. (fix #581) | If-else statement added to check if user is on macOS (semantic change) | Extracted dependency graphs and there are no changes. | After analyzing and comparing both versions of the outputs, the list of words do not provide enough information to infer any major changes. Individual words do not provide enough context to find any changes without refering back to the source code or change logs. |
| Base.java | In the previous version, Processing did not work properly for multiple users as the temp folder holding sketches was shared causing "auto naming limit" error message and stopped users to use software completely, even after restart the software. However, in this version, the class is now using UUID as names for temp folders to introduce another layer of indirection. UUID also prevents collisions when restarting the software. (fix #582, #549) | 1 parameter was added to the method ( semantic change) | No dependencies were found in this case. | In the new version of this file, "uuid" and "uuidrandomuuidtostring" words are added. This shows that, compared to the previous version, the uuid to replace naming was added. |

| | | | | |
|---|---|---|---|---|
| Github Project | https://github.com/archimatetool/archi | | | |
| | | SLOC | Documentation Link | |
| Version #1 | https://github.com/archimatetool/archi/releases/tag/release_4.9.3 | | | |
| Version #2 | https://github.com/archimatetool/archi/releases/tag/release_4.10.0 | | | |
| | | | | |
| | | | | |
| **Conceptual Changes** | | | | |
| | **Change Log** | | | |
| **What's fixed?** | | | | |
| **What's added?** | | | | |
| | | | | |
| Classes/Files Changed | Explanation | Change Type | Dependency Changes using CDA | TF - IDF Analysis |
| AbstractModelAction.java | New abstract class which adds to our existing hierarchy. Classes like ExportModelAction and ImportintoModelAction now extend this class instead of AbstractModelSelectionAction. The purpose of creating this new abstract class is to be suitable for menu items that update their enabled state when they are shown, but not suitable for toolbar items that don't actively update their state. | architectural change | *ExportModelAction dep AbstractModelSelection Action<br>- No longer dependant on it in v2 | Cannot extract the output csv files using TF-IDF because it is a newly added file. New files do not work with TF-IDF since we have no baseline version to compare the revised version with (it would be a blank output compared to the second version's output which is useless). |
| AbstractPaletteRoot | New abstract class that acts as the format painter tool entry created by sub-classes, it mainly creates a group of tools like a selection tool, marquee selection stack, and format painter. Subclasses like ArchimateDiagramEditorPalette have to implement the latter functionality now, making it a behavioral change/addition. | new functionality/behaviour | - We can't see function dependencies using CDA, and in this case the 2 versions are only different in the attributes/methods inside each subclass of AbstractPaletteRoot | After analyzing, the new version of this file added different tools, which includes selection tool, marquee selection stack and painter tools. |
| createToolsGroup / createControlsGroup | The createControlsGroup function inside ArchimateDiagramEditorPalette class got moved to the AbstractPaletteRoot and got renamed to createToolsGroup. We are simply moving the createToolsGroup function higher in our hierarchy and we now allow any class that extends the AbstractPaletteRoot class to have access to the function createToolsGroup. Since the ArchimateDiagramEditorPalette class extends the AbstractPaletteRoot class, it will still be able to use this function. | architectural change | - We can't see function dependencies using CDA, and in this case the 2 versions are only different in the attributes/methods inside each subclass of AbstractPaletteRoot | The createcontrolsgroup has been removed from the output file. In the output file of the AbstractPaletteRoot, the createToolsGroup was added in. This indicates that createControlsGroup has been removed but looking just at the analysis does not give enough information that it has changed to createCrontsGroup |
| getPaletteRoot (SketchEditor) | The getPaletteRoot function now returns a SketchEditorPalette object (previously PaletteRoot object). There is a change in dependency in the SketchEditor class so this is a semantic change. | semantic change | - We can't see function dependencies using CDA. | The PaletteRoot object was missing from the output file of the 4.10.0 version. This is a strong indication that the previous return type of getPaletteRoot function has changed to SketchEditorPalette. |
| clipBoardImageTransfer | A new helper class was created to transfer images from the clipboard. Previously, this functionality was done directly from where the file needed to be transfered. | architectural change | ExportAsImageToClipBoardAction is now depends on clipBoardImageTransfer instead of depending PlatformUtils directly | Cannot extract the output csv files using TF-IDF because it is a newly added file. New files do not work with TF-IDF since we have no baseline version to compare the revised version with (it would be a blank output compared to the second version's output which is useless). |

| Repo | >2 Revisions | Contributors | Framework | Has strong indep. documentation? | Votes | |
|---|---|---|---|---|---|---|
| https://github.com/jagrosh/MusicBot | Yes | 12 | No | Github Changelogs and Wiki<br>https://jmusicbot.com/ | Valerie, Hao, Eddie, Ahmad | |
| https://github.com/facebook/fresco | yes | 217 | No | Yes<br>https://frescolib.org/docs/building-from-source.html | | |
| https://github.com/Anuken/Mindustry/releases | yes | 491 | no | yes? | | |
| https://github.com/shashirajraja/onlinebookstore | yes | 27 | no | no | | |
| https://github.com/jagrosh/GiveawayBot | Yes | 8 | No | No | | |
| https://github.com/processing/processing4 | yes | 136 | yes | yes | | |
| https://github.com/Zettlr/Zettlr | yes | 81 | no | no | | |
| https://github.com/frappe/erpnext | yes | 443 | yes | yes | | |
| https://github.com/pypa/pipenv | yes | 390 | no | yes | | |
| https://github.com/sonnyp/Workbench | yes | 36 | yes | | | |
| https://github.com/klembot/twinejs | yes | 38 | no | yes | | |
| https://github.com/gnome-terminator/terminator | yes | 87 | yes | yes | | |
| https://github.com/shashirajraja/onlinebookstore | yes | 27 | yes? | yes | | |
| https://github.com/PhilJay/MPAndroidChart | yes | 69 | yes? | yes? | | |