Big Commit Analysis

Manual investigation of possible big commits

Using the example project Lucene-Solr

General

Many possible big commits like LUCENE-3892, LUCENE-3312, LUCENE-3846, LUCENE-5339, LUCENE-3069, LUCENE-4055, LUCENE-5487, etc. have a commit message similar to “merge trunk” or “merge master”. The issue report itself does not have a merge type, instead just reflects the latest commit that was merged within this commit. These are obviously big commits because they merge several changes.

LUCENE-5487

There are about 18 commits with this issue identified. The issue tracker’s message is “Can we separate ‘top scorer’ from ‘sub scorer’?” and a patch is attached. And one of these commits has the message “merge trunk” which contains many changes.

LUCENE-6271

This issue is classified as a bug, but actually there are 42 commits assigned to this issue, where 3 are trunk merges. The biggest merge commit (Ref: 05cf3fde0d909a492df8e82c119c4b632defc709) of this issue has 6700 changes and 1100 files touched.

LUCENE-3079

The linked commit has 40433 additions and 263 files changed. This one adds a whole new module called “facet” to the project that was probably developed aside of the project. This is not just simply a new feature or an improvement.

LUCENE-5882

This is another commit with 59 files touched and 3071 changes (additions + deletions). This updates some documentation, but it’s categorized as improvement. In Apache projects, mostly documentation commits are categorized as improvements as observed in the developing process.

SOLR-9083

This one is categorized as improvement, has 28500 changes and 150 touched files. It removes deprecated “types” and “fields” from xml schemas and sometimes, there are just spaces removed to possibly fit a code convention. So this is in our opinion not a big commit.

LUCENE-5468, Ref 2e0fc562bc239ea897023796160a8870eddd2a48

This commit is classified as a bug, the message just says “commit current state” but there are only new files added. This does not look like a bug fix. The issue itself has 15 commits linked.

This list is not concluding.

Possible ways to split big commits

LUCENE-6271, Ref 05cf3fde0d909a492df8e82c119c4b632defc709

Investigating the CHANGES.txt file (https://fisheye6.atlassian.com/changelog/Lucene?cs=1670257) gives us a list of some bug fixes, improvements and optimizations with a linked issue identifier. We could further analyze these issues and the related commits to have some file lists for every change and then iteratively remove all files from the big commit that belong to one of these issues.

Other commits

Perhaps with a model with good examples of bug fixes, improvements and features, we could be able to classify parts of the commit. It gets difficult when files in a big commit belong to different categories (e.g. File.java is touched by a bug fix and also by a new feature).