How to Bisect bug to a single patch

By Ezinne Nnamani

Collabora Online

Consultant Associate Productivity Software Engineer

nnamani.ezinne@collabora.com

Bisect/Bibisect

Regression

- A type of software bug where a feature that has worked before stops working.

Bisect

- Bisect is a process of finding a particular commit that caused regression by comparing the middlemost commit in a given range.
- Bibisect is a similar procedure but there are repositories of binaries used to find the commit that introduced the regression.
- It is required to find the good working state of the feature to be able to bisect the bug/regression. It can be done by checking the older versions.
- At each step of the bisect process, one would need to repeat the procedure that triggers the bug.

Testing using CODE Docker

Versions to test:

- The Docker images exist for each of the released versions.
- It is necessary to test with versions older than 4.0.
- There are lots of changes and improvements in the Collabora Online since 4.0 version.

To Test using CODE Docker:

- Test for the bug using a CODE Docker version.
- If the bug exists in that version, also check in an older version to see if the bug exists there.
- If the bug started in one version, for instance, within 6.4 updates, then take the two pairs or range of commit hashes to bisect on.
- If the bug started between different versions there will be a wider range of commit hashes to bisect with.

Understanding the Relationship between Core Build and Online build

- It is necessary that CollaboraOffice core is built for Collabora Online to run successfully.
- An Online build from any given time is composed of builds from the CollaboraOffice core and Collabora Online commits that were the latest at that time.
- CollaboraOffice core and Collabora Online builds often possibly months apart can be matched.
- When looking to find a commit causing regression, often only one of the repositories matter. Bisecting of the bug can be either using Online or using Core at a time.

Regression in CollaboraOffice CORE

- If the regression occurs in CollaboraOffice core where the Collabora Online is built, then the core can be bibisected.

- There are binary repositories for tracing regression and conduction bibisection in LibreOffice core. It has been used for tracing regression for a long time.

- Similar binary repositories also exist for Collabora branches.

- An example:

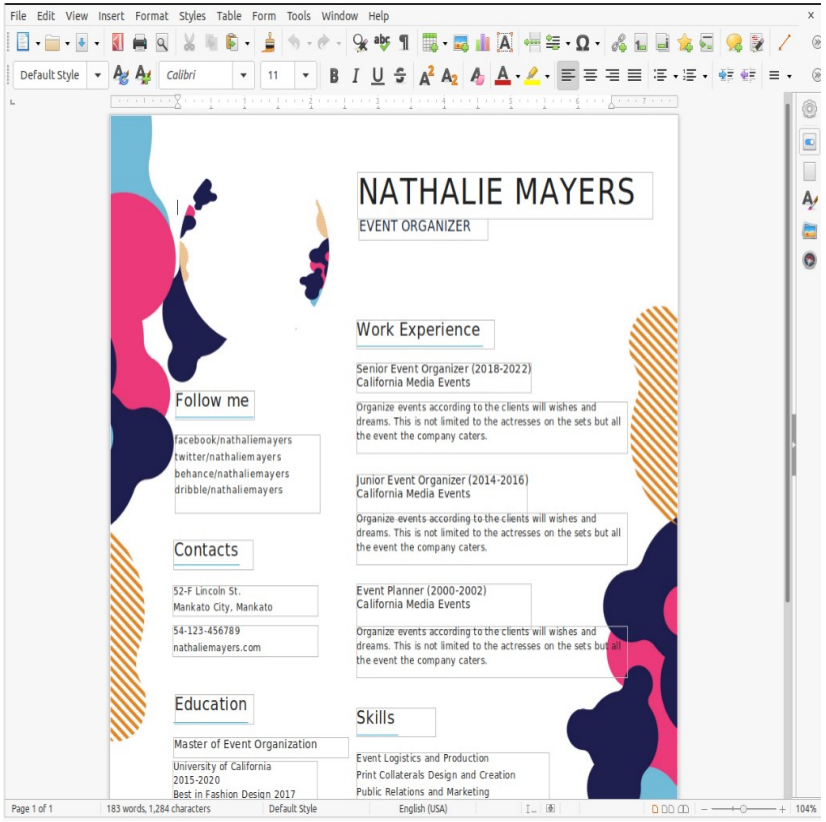
  . Image missing in DOCX file when opened.

  This regression has been fixed.

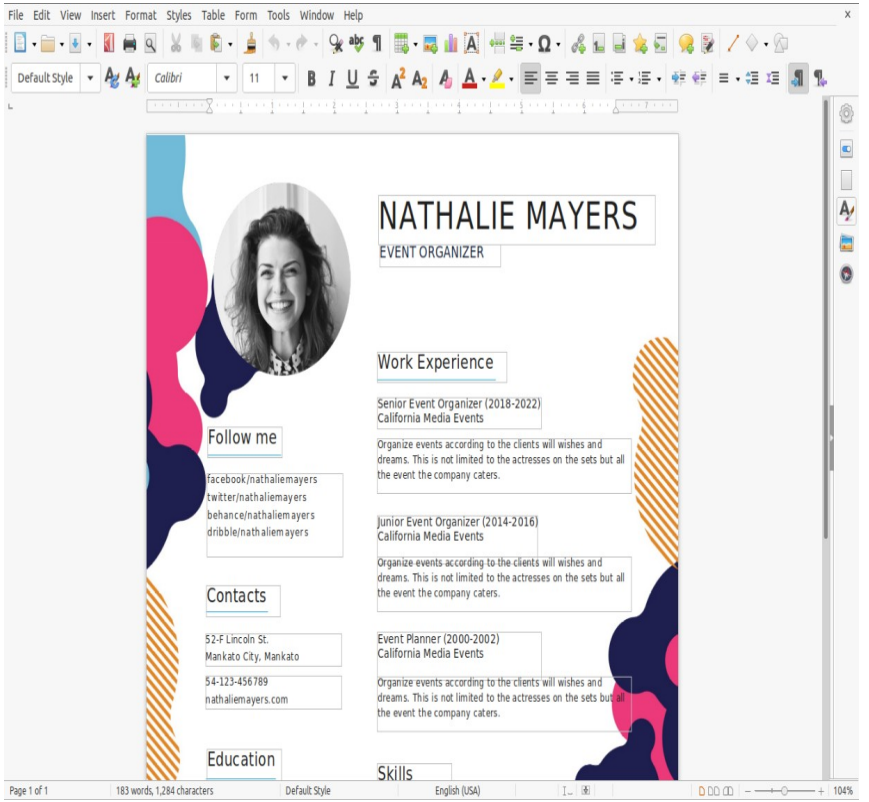
2021 - All about COOL days Collabora Online

Screenshot showing the regression of missing Image in DOCX on Collabora Core

With the Regression



Regression Fixed



Collabora Online

Regressions in Collabora Online repo

Regression in Online Desktop and Mobile:

- Recently, daily builds of APK for tracing the good state of the feature in mobile.
- No binary repo for bisecting in Online.

Example of regression in Online repo:

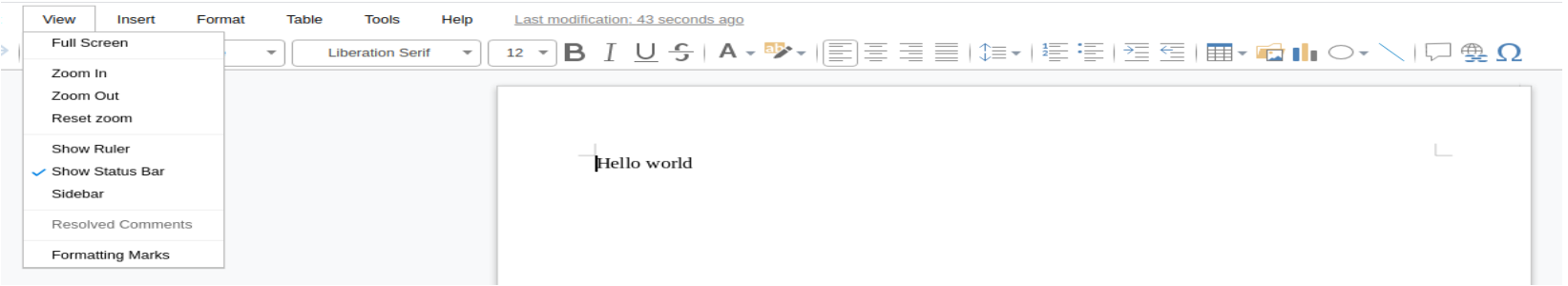
- Recently, the Show Ruler feature in Writer stopped working. We remember using the Writer Ruler about the month ago and it worked well.

commit : 2cad88e8bf0bbcc888734d0fdf75cf824a4ce5ed

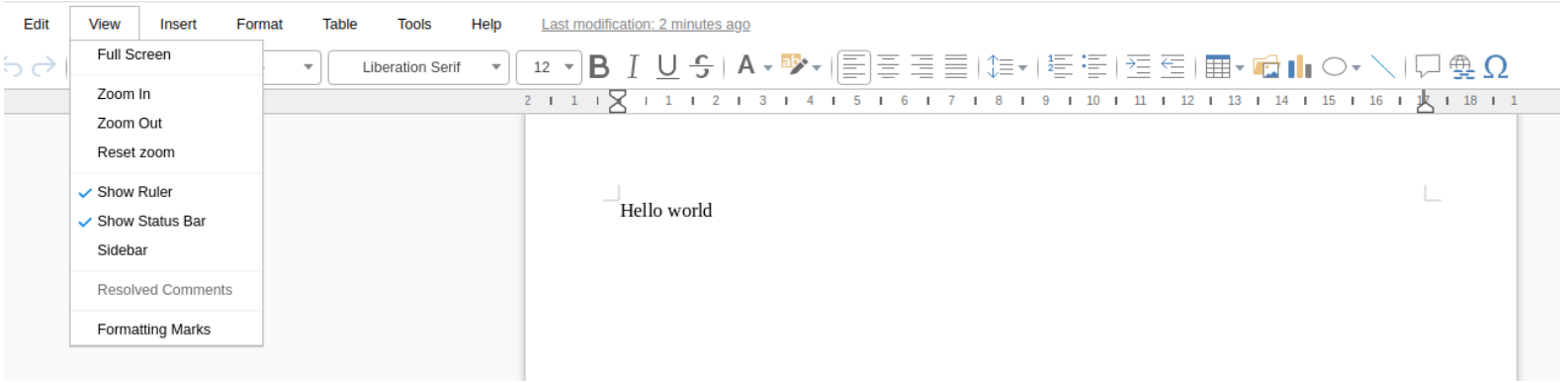
commit message: Notebookbar: Center save icon

Screenshots showing the Show Ruler Regression in Online

With Regression:



Regression Fixed:



Bisection in Collabora Online repo

Steps in bisection using Online repo.

•The previous example was a regression and to bisect it, the following steps were taken:

1. The current commit where the bug exists will be referred to as Newer commit while the commit of the good state will be refered to as Older commit.

2. Use the command git bisect start <newer commit> <older commit>.

3. Compile build in each commit checked out to.
   - To avoid errors during compilation:
     * Use “–disable-werror” in the ./configure parameter
     * Add “*.js” in the loleaflet/.eslint file to avoid JS linting errors.

4. Test for the bug in each build.
   -If bug exists, terminate the build and enter the command git bisect bad. Otherwise, enter git bisect good

5. After about 7 to 8 builds, the commit causing regression will be obtained;
   In the test case, we have the result as

bdeb7154305fe7859a35a5b53ebb2b8bcf301207 is the first bad commit

commit:	bdeb7154305fe7859a35a5b53ebb2b8bcf301207
commit message:	Readonly mode: Do not display ruler

2021 - All about

COOL days

Collabora Online

Thanks !

Collabora Online

By Ezinne Nnamani

@CollaboraOffice
hello@collaboraoffice.com
Collaboraoffice.com