

RQ2: Type Inconsistency

This experiment is in the **Section V-C Inconsistency of Binding Calls** of our paper. The goal of this experiment is to

1. compare the type information of Adobe Reader inferred by TYPEORACLE with the type information inferred by Adobe API Manual;
2. compare the type information of Adobe Reader inferred by TYPEORACLE with the type information of Foxit Reader inferred by TYPEORACLE.

The result is shown in the **Fig. 6: Systematic study of binding call inconsistency** in our paper. We can see from Figure 6a that only 53% of Adobe Reader's binding calls are documented, 21% of which have type inconsistency with the documentation. Further inspection reveals that the inconsistency comes from ambiguous description and incomplete parameter list in the documentation. Figure 6b says that Foxit Reader shares 42% of binding calls with Adobe Reader, 36% of which differ in either the number of parameters or the type of certain parameters.

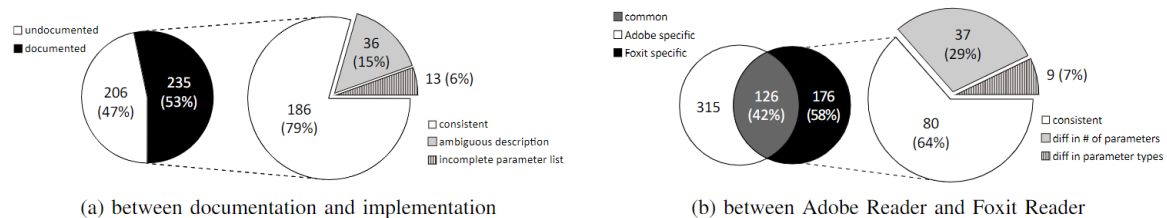


Fig. 6: Systematic study of binding call inconsistency.

folder structure

This following are the file names and their descriptions.

`cmp_adobe_foxit`: compare the type information of Adobe with the type information of Foxit

`cmp_doc`: compare the type information of TypeOracle with the type information of Manual

`compare_api`:

- `cmp_adobe_foxit_api.py`: script to get foxit specific APIs and common APIs

`adobe_doc.txt`: APIs present in Adobe API Manual

`adobe_undoc.txt`: APIs not present in Adobe API Manual

`difference_adobe_foxit.txt`: the different APIs in Adobe Reader and Foxit Reader, including APIs that are different in parameters and APIs that are different in parameters' types

`difference_manual.txt`: the different APIs in TypeOracle and Adobe Manual in Adobe's API list, including APIs that have ambiguous description and incomplete parameter

foxit_specific_api.txt: APIs that only implemented in Foxit Reader

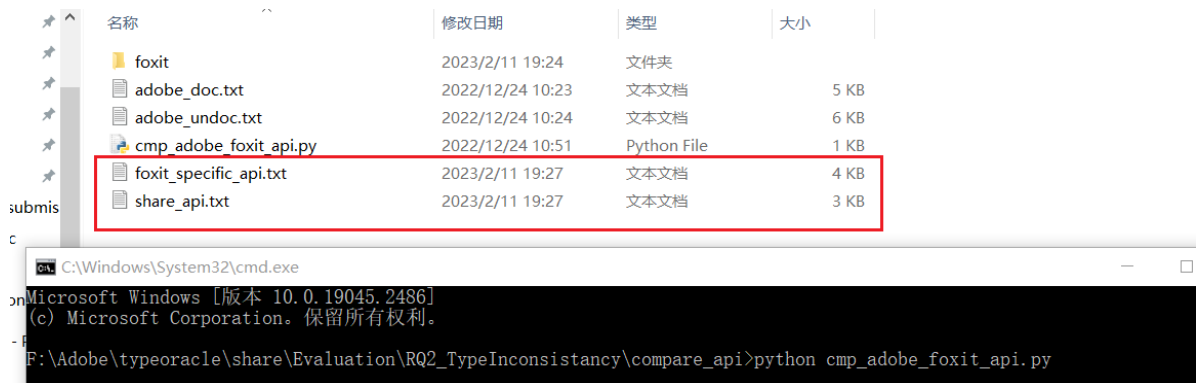
share_api.txt: APIs that both implemented in Adobe Reader and Foxit Reader

How to reproduce

1. prepare these files: adobe_doc.txt, adobe_undoc.txt, func1st.txt(foxit), setter1st.txt(foxit)

foxit	2023/2/11 19:24	文件夹	
adobe_doc.txt	2022/12/24 10:23	文本文档	5 KB
adobe_undoc.txt	2022/12/24 10:24	文本文档	6 KB
cmp_adobe_foxit_api.py	2022/12/24 10:51	Python File	1 KB

2. execute `cmp_adobe_foxit_api.py` to get foxit specific APIs and common APIs



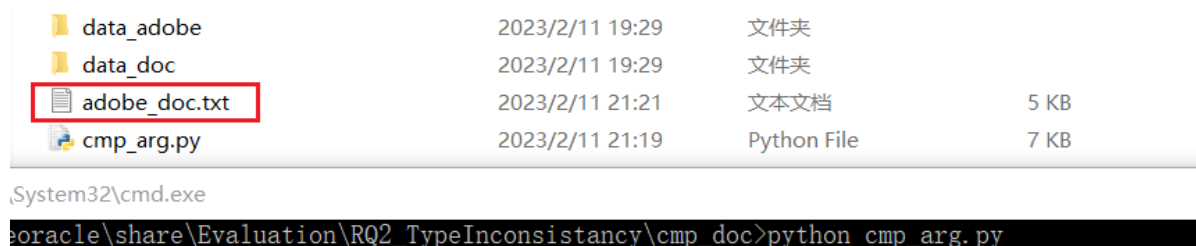
名称	修改日期	类型	大小
foxit	2023/2/11 19:24	文件夹	
adobe_doc.txt	2022/12/24 10:23	文本文档	5 KB
adobe_undoc.txt	2022/12/24 10:24	文本文档	6 KB
cmp_adobe_foxit_api.py	2022/12/24 10:51	Python File	1 KB
foxit_specific_api.txt	2023/2/11 19:27	文本文档	4 KB
share_api.txt	2023/2/11 19:27	文本文档	3 KB

```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.19045.2486]
(c) Microsoft Corporation。保留所有权利。
F:\Adobe\typeoracle\share\Evaluation\RQ2_TypeInconsistency\compare_api>python cmp_adobe_foxit_api.py
```

3. compare the type information of TypeOracle with the type information of Manual

data_adobe	2023/2/11 19:29	文件夹	
data_doc	2023/2/11 19:29	文件夹	
cmp_arg.py	2023/2/11 21:19	Python File	7 KB

4. execute `cmp_arg.py` to get result in `adobe_doc.txt`



名称	修改日期	类型	大小
data_adobe	2023/2/11 19:29	文件夹	
data_doc	2023/2/11 19:29	文件夹	
adobe_doc.txt	2023/2/11 21:21	文本文档	5 KB
cmp_arg.py	2023/2/11 21:19	Python File	7 KB

```
C:\Windows\System32\cmd.exe
F:\Adobe\typeoracle\share\Evaluation\RQ2_TypeInconsistency\cmp_doc>python cmp_arg.py
```

5. compare the type information of Adobe with the type information of Foxit

data_adobe	2023/2/11 19:29	文件夹	
data_foxit	2023/2/11 19:29	文件夹	
cmp_arg.py	2023/2/11 21:19	Python File	7 KB

6. execute cmp_arg.py to get result in `adobe_foxit.txt`

data_adobe	2023/2/11 19:29	文件夹	
data_foxit	2023/2/11 19:29	文件夹	
adobe_foxit.txt	2023/2/11 21:24	文本文档	7 KB
cmp_arg.py	2023/2/11 21:19	Python File	7 KB

ws\System32\cmd.exe

```
ypeoracle\share\Evaluation\RQ2_TypeInconsistency\cmp_adobe_foxit>python cmp_arg.py  
queryServices  
resolveService
```