WITH YourTable AS( SELECT 'main.c' as difference\_old\_path

UNION ALL SELECT 'main.cpp'

UNION ALL SELECT 'thefile.java'

)

SELECT lower(substr(difference\_old\_path, 0

, length(difference\_old\_path) - strpos(reverse(difference\_old\_path),'.') ))

, length(difference\_old\_path) as len

, strpos(reverse(difference\_old\_path),'.') as point\_pos

FROM YourTable

# Standard Sql, large results

# into abstarction\_related\_commits

Select \*

From

`citric-trees-187317.repo\_attributes.valid\_commits\_classified`

Where

# date after 2008, GitHub funding year (avoid old /possibly corrupted data)

TIMESTAMP\_DIFF(author\_date, cast( '2008-01-01' as TIMESTAMP), DAY) > 0

# date until 2019 to ease reproducibility

and

TIMESTAMP\_DIFF( author\_date, cast( '2019-01-01' as TIMESTAMP), DAY) < 0

And

lower(reverse(substr(reverse(difference\_old\_path), 0, strpos(reverse(difference\_old\_path),'.')))) in ('.c', '.cc', '.cpp', '.cxx', '.c++', '.h', '.hh', '.hpp', '.hxx', '.h++')

# Standard Sql, large results

# into commits\_interface

Select

Repo\_name

, commit

, lower(substr(difference\_old\_path, 0

, length(difference\_old\_path) - strpos(reverse(difference\_old\_path),'.') )) as file\_stem

, max(author\_date) as author\_date

, max((LENGTH(REGEXP\_REPLACE(lower(difference\_old\_path), 'test', '#'))

- LENGTH(REGEXP\_REPLACE(lower(difference\_old\_path), 'test', ''))) > 0) as is\_test

, max(lower(reverse(substr(reverse(difference\_old\_path), 0, strpos(reverse(difference\_old\_path),'.')))) in ('.c', '.cc', '.cpp', '.cxx', '.c++')

) as has\_imp\_file

, max(lower(reverse(substr(reverse(difference\_old\_path), 0, strpos(reverse(difference\_old\_path),'.')))) in ( '.h', '.hh', '.hpp', '.hxx', '.h++')) as has\_interface\_file

From

`citric-trees-187317.repo\_attributes.abstarction\_related\_commits`

Where

# date after 2008, GitHub funding year (avoid old /possibly corrupted data)

TIMESTAMP\_DIFF(author\_date, cast( '2008-01-01' as TIMESTAMP), DAY) > 0

# date until 2019 to ease reproducibility

and

TIMESTAMP\_DIFF( author\_date, cast( '2019-01-01' as TIMESTAMP), DAY) < 0

Group by

Repo\_name

, file\_stem

, commit

# Abstraction level dist

Select

round(abstraction,2) as rabstraction

, count(\*) as files

from

(

Select

Repo\_name

, file\_stem

, if(Sum(if(has\_imp\_file, 1,0)) > 0

,1.0\*Sum(if(has\_imp\_file and has\_interface\_file, 1,0))/Sum(if(has\_imp\_file, 1,0))

, 0) as abstraction

From

`citric-trees-187317.repo\_attributes.commits\_interface`

Group by

Repo\_name

, file\_stem

Having count(\*) > 9

) as InSql

Group by

Rabstraction

Order by

rabstraction

# Compute refactor abstraction

# Standard, large results

# into refactor\_abstraction\_stats

Select

Refs.refactor\_repo\_name as repo\_name

, refs.refactor\_commit as commit

, max(bug\_ratio\_after\_3\_months) as bug\_ratio\_after\_3\_months

, max(bug\_ratio\_before\_3\_months) as bug\_ratio\_before\_3\_months

, max(bug\_ratio\_after\_6\_months) as bug\_ratio\_after\_6\_months

, max(bug\_ratio\_before\_6\_months) as bug\_ratio\_before\_6\_months

, max(commits\_after\_3\_months) as commits\_after\_3\_months

, max(commits\_before\_3\_months) as commits\_before\_3\_months

, max(commits\_after\_6\_months) as commits\_after\_6\_months

, max(commits\_before\_6\_months) as commits\_before\_6\_months

, if(Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < 3\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > 1 and has\_imp\_file, 1,0)) > 0

,1.0\*Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < 3\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > 1 and has\_imp\_file and has\_interface\_file, 1,0))/Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < 3\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > 1 and has\_imp\_file, 1,0))

, 0) as abstraction\_after\_3\_months

, if(Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > - 3\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < -1

and has\_imp\_file, 1,0)) > 0

,1.0\*Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > - 3\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < -1

and has\_imp\_file and has\_interface\_file, 1,0))/Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > - 3\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < -1

and has\_imp\_file, 1,0))

, 0) as abstraction\_before\_3\_months

, if(Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < 6\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > 1 and has\_imp\_file, 1,0)) > 0

,1.0\*Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < 6\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > 1 and has\_imp\_file and has\_interface\_file, 1,0))/Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < 6\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > 1 and has\_imp\_file, 1,0))

, 0) as abstraction\_after\_6\_months

, if(Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > - 6\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < -1

and has\_imp\_file, 1,0)) > 0

,1.0\*Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > - 6\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < -1

and has\_imp\_file and has\_interface\_file, 1,0))/Sum(if(TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) > - 6\*30 and TIMESTAMP\_DIFF(interface.author\_date, refctor\_date, DAY) < -1

and has\_imp\_file, 1,0))

, 0) as abstraction\_before\_6\_months

from

`citric-trees-187317.repo\_attributes.refactor\_stats` as refs

Join

`citric-trees-187317.repo\_attributes.commits\_interface` as interface

on

# Join on stem

Interface.file\_stem = lower(substr(refs.file, 0

, length(refs.file) - strpos(reverse(refs.file),'.') ))

where

# C/CPP implementation files

Refs.refactor\_file\_extension in ('.c', '.cc', '.cpp', '.cxx', '.c++')

Group by

Repo\_name

, commit

#### Abstraction results

Select

count(\*) as refactors

, avg(bug\_ratio\_after\_6\_months -bug\_ratio\_before\_6\_months) as diff\_6\_month

, avg(bug\_ratio\_after\_3\_months -bug\_ratio\_before\_3\_months) as diff\_3\_month

, sum(if (abstraction\_after\_3\_months - abstraction\_before\_3\_months < - 0.1,1,0)) as abstraction\_1\_refactors

, avg(if (abstraction\_after\_3\_months - abstraction\_before\_3\_months < - 0.1 , bug\_ratio\_after\_3\_months -bug\_ratio\_before\_3\_months, null)) as abs\_diff\_3\_month

, avg(if (abstraction\_after\_6\_months - abstraction\_before\_6\_months < - 0.1 , bug\_ratio\_after\_6\_months -bug\_ratio\_before\_6\_months, null)) as abs\_diff\_6\_month

, sum(if (abstraction\_after\_3\_months - abstraction\_before\_3\_months < - 0.1

And commits\_after\_3\_months > 4 and commits\_after\_3\_months > 4,1,0)) as abs\_commits\_4\_refactors

, avg(if (abstraction\_after\_3\_months - abstraction\_before\_3\_months < - 0.1

And commits\_after\_3\_months > 4 and commits\_after\_3\_months > 4 , bug\_ratio\_after\_3\_months -bug\_ratio\_before\_3\_months, null)) as abs\_commits\_4\_diff\_3\_month

, avg(if (abstraction\_after\_6\_months - abstraction\_before\_6\_months < - 0.1

And commits\_after\_6\_months > 4 and commits\_after\_6\_months > 4 , bug\_ratio\_after\_6\_months -bug\_ratio\_before\_6\_months, null)) as abs\_commits\_4\_diff\_6\_month

from

`citric-trees-187317.repo\_attributes.refactor\_abstraction\_stats`

# abstraction/ccp cm on clean

Select

Bug\_ratio\_after\_3\_months < bug\_ratio\_before\_3\_months as improved\_ccp

,abstraction\_after\_3\_months < abstraction\_before\_3\_months as improved\_abstraction

, count(\*) as cases

from

[citric-trees-187317:repo\_attributes.refactor\_abstraction\_stats] as s

join

[citric-trees-187317:repo\_attributes.refactor\_stats\_2018\_commit\_size] as cs

#join rule

on

s.repo\_name = cs.refactor\_repo\_name

and

s.commit = cs.refactor\_commit

Where

(commits\_before\_3\_months > 9)

and (commits\_after\_3\_months > 9)

#and (refactor\_before\_3\_months =0)

#and (refactor\_after\_3\_months =0)

and (cs.non\_test\_files =1)

Group by

Improved\_ccp

, improved\_abstraction

Order by

Improved\_ccp

, improved\_abstraction