

# Alambic – R analysis document

*Boris Baldassari*

## Contents

<b>Summary</b>	<b>3</b>
<b>Metrics</b>	<b>4</b>
SCM Commits one month . . . . .	5
Project committers one year . . . . .	5
SCM authors one week . . . . .	5
Public documented API (%) . . . . .	5
Number of critical issues . . . . .	5
SCM information . . . . .	6
SCM committers one year . . . . .	6
User ML Posts . . . . .	6
ITS information . . . . .	6
ITS issues created last week . . . . .	6
Technical debt . . . . .	7
SCM Commits one year . . . . .	7
ITS Open issues . . . . .	7
SCM Changed Lines . . . . .	7
Doc information . . . . .	7
Number of green jobs . . . . .	7
SCM Changed Lines one week . . . . .	8
User ML Threads . . . . .	8
User ML Authors . . . . .	8
Stack Overflow Views (5Y) . . . . .	8
Number of jobs . . . . .	8
Package Tangle index . . . . .	9
Number of red jobs . . . . .	9
User ML Authors . . . . .	9
Stack Overflow Askers (5Y) . . . . .	9
Project authors one month . . . . .	9
Public API . . . . .	9
Number of functions . . . . .	10
SCM Still Open Pull requests one year . . . . .	10
User ML Posts . . . . .	10
User ML Threads . . . . .	10
SCM Open Pull requests one year . . . . .	10
User ML Threads . . . . .	11
Number of yellow jobs . . . . .	11
Number of stars . . . . .	11
ITS Open issues (%) . . . . .	11
SCM authors one year . . . . .	11
File complexity . . . . .	12
SCM authors . . . . .	12
Stack Overflow Votes (5Y) . . . . .	12
Number of failed jobs one week . . . . .	12
Project Commits one year . . . . .	12
SCM Staled Open Pull requests one month . . . . .	13

SCM Open Pull requests one week . . . . .	13
Project Commits . . . . .	13
CI information . . . . .	13
Project authors one week . . . . .	13
Last activity . . . . .	14
Test coverage . . . . .	14
Duplicated lines (%) . . . . .	14
SCM Merged Pull requests . . . . .	14
SCM Closed Pull requests . . . . .	14
Stack Overflow Questions (5Y) . . . . .	14
Number of minor issues . . . . .	15
Project committers . . . . .	15
Sqale Debt ratio . . . . .	15
ITS issues created last year . . . . .	15
SCM committers . . . . .	15
User ML Authors . . . . .	15
ITS authors last year . . . . .	16
ITS issues created last month . . . . .	16
Maintainability rating . . . . .	16
ITS Authors . . . . .	16
ITS issues updated last month . . . . .	16
User ML Threads . . . . .	17
ITS Late issues . . . . .	17
ITS Pending issues . . . . .	17
Project committers one month . . . . .	17
Project authors one year . . . . .	17
User ML Posts . . . . .	18
Open issues . . . . .	18
ITS authors last month . . . . .	18
ITS issues updated last year . . . . .	18
Number of forks . . . . .	18
Ratio of green jobs . . . . .	18
Project committers one week . . . . .	19
Branch coverage . . . . .	19
Access information . . . . .	19
Project authors . . . . .	19
Comment lines density . . . . .	19
Number of comment lines . . . . .	20
Number of major issues . . . . .	20
Number of lines of code . . . . .	20
Number of releases . . . . .	20
SCM Changed Lines one year . . . . .	20
SCM Pull requests . . . . .	21
SCM authors one month . . . . .	21
SCM Still Open Pull requests one week . . . . .	21
ITS issues updated last week . . . . .	21
Commented code . . . . .	21
Project Commits one month . . . . .	22
SCM Commits one week . . . . .	22
SCM Changed Lines one month . . . . .	22
Stack Overflow Answer rate (5Y) . . . . .	22
SCM Open Pull requests . . . . .	22
Number of files . . . . .	23
ITS Total issues . . . . .	23

SCM Commits . . . . .	23
SCM committers one week . . . . .	23
Number of info issues . . . . .	23
User ML Posts . . . . .	23
Line coverage . . . . .	24
SCM Still Open Pull requests one month . . . . .	24
Number of releases . . . . .	24
Stack Overflow Answers (5Y) . . . . .	24
Project Commits one week . . . . .	24
SCM Open Pull requests one month . . . . .	25
ITS authors last week . . . . .	25
Total complexity . . . . .	25
SCM committers one month . . . . .	25
User ML Authors . . . . .	25
Number of statements . . . . .	26
Number of blocker issues . . . . .	26
<b>Attributes</b>	<b>26</b>
Activity . . . . .	26
Diversity . . . . .	26
Documentation . . . . .	27
Ecosystem . . . . .	27
Maintainability . . . . .	27
Process . . . . .	27
Product . . . . .	27
Eclipse Maturity . . . . .	28
Reliability . . . . .	28
Build and Release Management . . . . .	28
Configuration Management . . . . .	28
Support . . . . .	28
<b>Git analysis</b>	<b>29</b>
Weekly commits . . . . .	29
Weekly authors . . . . .	30

## Summary

This plugin generates a PDF document with information about project **modeling.mdt.papyrus**.

This plugin is intended as an example of R Markdown document to help people easily setup their own R analysis on software development data.

# Metrics

Mnemo	Value
CI_JOBS	121.00
CI_JOBS_FAILED_1W	4.00
CI_JOBS_GREEN	93.00
CI_JOBS_GREEN_RATIO	92.00
CI_JOBS_RED	4.00
CI_JOBS_YELLOW	4.00
ITS_AUTHORS	493.00
ITS_AUTHORS_1M	9.00
ITS_AUTHORS_1W	5.00
ITS_AUTHORS_1Y	44.00
ITS_CREATED_1M	31.00
ITS_CREATED_1W	10.00
ITS_CREATED_1Y	524.00
ITS_ISSUES_ALL	11875.00
ITS_OPEN	2519.00
ITS_OPEN_OLD	0.00
ITS_OPEN_PERCENT	21.00
ITS_OPEN_UNASSIGNED	0.00
ITS_UPDATED_1M	31.00
ITS_UPDATED_1W	10.00
ITS_UPDATED_1Y	524.00
MLS_USR_AUTHORS	1254.00
MLS_USR_AUTHORS_1M	24.00
MLS_USR_AUTHORS_1W	6.00
MLS_USR_AUTHORS_1Y	133.00
MLS_USR_POSTS	2609.00
MLS_USR_POSTS_1M	80.00
MLS_USR_POSTS_1W	14.00
MLS_USR_POSTS_1Y	690.00
MLS_USR_THREADS	2609.00
MLS_USR_THREADS_1M	33.00
MLS_USR_THREADS_1W	7.00
MLS_USR_THREADS_1Y	191.00
PROJECT_ACCESS_INFO	3.00
PROJECT_DOC_INFO	5.00
PROJECT_ITS_INFO	5.00
PROJECT_REL_VOL	32.00
PROJECT_SCM_INFO	1.00
SCM_AUTHORS	121.00
SCM_AUTHORS_1M	5.00
SCM_AUTHORS_1W	3.00
SCM_AUTHORS_1Y	15.00
SCM_COMMITS	14845.00
SCM_COMMITS_1M	32.00
SCM_COMMITS_1W	9.00
SCM_COMMITS_1Y	306.00
SCM_COMMITTERS	98.00
SCM_COMMITTERS_1M	5.00
SCM_COMMITTERS_1W	3.00
SCM_COMMITTERS_1Y	14.00
SCM_MOD_LINES	77052570.00
SCM_MOD_LINES_1M	14831.00
SCM_MOD_LINES_1W	6939.00
SCM_MOD_LINES_1Y	860817.00
SO_ANSWERS_VOL_5Y	61.00
SO_ANSWER_RATE_5Y	0.87
SO_ASKERS_5Y	60.00
SO_QUESTIONS_VOL_5Y	70.00
SO_VIEWS_VOL_5Y	24691.00
SO_VOTES_VOL_5Y	67.00
SQ_COMMENT_LINES	122797.00
SQ_COMR	27.60
SQ_CPX	75102.00
SQ_CPX_CLASS_IDX	15.50
SQ_CPX_FILE_IDX	18.00
SQ_CPX_FUNC_IDX	2.60
SQ_DUPLICATED_BLOCKS	963.00
SQ_DUPLICATED_FILES	408.00
SQ_DUPLICATED_LINES	41324.00
SQ_DUPLICATED_LINES_DENSITY	6.30
SQ_FILES	4173.00
SQ_FUNCS	28649.00
SQ_ISSUES_OPEN	19611.00
SQ_NCLOC	321906.00
SQ_NCLOC_LANG	
SQ_PACKAGES_CYCLES	457.00
SQ_PACKAGES_TANGLE_IDX	21.90
SQ_PUBLIC_API	17126.00
SQ_PUBLIC_API_DOC_DENSITY	74.80
SQ_PUBLIC_UNDOC_API	4316.00
SQ_RULES	35.00
SQ_SQALE_DEBT_RATIO	3.90
SQ_SQALE_INDEX	379145.00
SQ_SQALE_RATING	1.00
SQ_STATEMENTS	135946.00
SQ_VIOLATIONS	19611.00

## SCM Commits one month

ID: SCM\_COMMITS\_1M

Value: 32 ( 4 / 5 )

Description: Total number of commits in source code management repositories dated during the last month. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## Project committers one year

ID: PROJECT\_COMMITTERS\_1Y

Value:

Description: Total number of identities found as committers of commits in source code management repositories dated during the last year. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'committer date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## SCM authors one week

ID: SCM\_AUTHORS\_1W

Value: 3

Description: Total number of identities found as authors of commits in source code management repositories dated during the last week. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.

## Public documented API (%)

ID: SQ\_PUBLIC\_API\_DOC\_DENSITY

Value: 74.8

Description: Density of public documented API = (Public API - Public undocumented API) / Public API \* 100

## Number of critical issues

ID: SQ\_VIOLATIONS\_CRITICAL

Value: 616

Description: The total number of issues (violations) found by SonarQube with a severity equal to CRITICAL.

## SCM information

ID: PROJECT\_SCM\_INFO

Value: 1 ( 4 / 5 )

Description: Is the source\_repo info correctly filled in the PMI records? The project management infrastructure file holds information about one or more source repositories. This test checks that at least one source repository is defined, and accessible.

## SCM committers one year

ID: SCM\_COMMITTERS\_1Y

Value: 14

Description: Total number of identities found as committers of commits in source code management repositories dated during the last year. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'committer date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## User ML Posts

ID: MLS\_USR\_POSTS\_1M

Value: 80 ( 5 / 5 )

Description: The total number of posts found in the User mailing list during last month. Having many posts shows the mailing list is active. It encourages people to participate, ask and answer questions.

## ITS information

ID: PROJECT\_ITS\_INFO

Value: 5 ( 5 / 5 )

Description: Is the bugzilla info correctly filled in the PMI records? The project management infrastructure file holds information about one or more bugzilla instances. This test checks that at least one bugzilla instance is defined, with a product identifier, a create\_url to enter a new issue, and a query\_url to fetch all the issues for the project.

## ITS issues created last week

ID: ITS\_CREATED\_1W

Value: 10

Description: Number of issues created during last week. If today is Wed. 2017-02-01 then the range is from Wed. 2017-01-25 to Wed. 2017-02-01.

## Technical debt

ID: SQ\_SQALE\_INDEX

Value: 379145

Description: Effort to fix all maintainability issues. The measure is stored in minutes in the DB.

## SCM Commits one year

ID: SCM\_COMMITS\_1Y

Value: 306

Description: Total number of commits in source code management repositories dated during the last year. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## ITS Open issues

ID: ITS\_OPEN

Value: 2519 ( 5 / 5 )

Description: Number of issues with a state 'open' at the time of analysis.

## SCM Changed Lines

ID: SCM\_MOD\_LINES

Value: 77052570

Description: Total number of changed lines (added, removed, changed) in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date).

## Doc information

ID: PROJECT\_DOC\_INFO

Value: 5 ( 4 / 5 )

Description: Is the documentation info correctly filled in the PMI records? The project management infrastructure file holds information about various documentation and manuals. This test checks the number of doc-related entries defined in the PMI: build\_doc, documentation, documentation\_url, forums, gettingstarted\_url, mailing\_lists, website\_url, wiki\_url.

## Number of green jobs

ID: CI\_JOBS\_GREEN

Value: 93

Description: The number of green (successful) jobs on the Hudson engine. Green (or blue) jobs in Hudson define successful builds.

## **SCM Changed Lines one week**

ID: SCM\_MOD\_LINES\_1W

Value: 6939

Description: Total number of changed lines (added, removed, changed) in source code management repositories dated during the last week. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.

## **User ML Threads**

ID: MLS\_USR\_THREADS

Value: 2609

Description: The total number of threads (one question followed by zero or more answers) found in the User mailing list. Having many threads shows the mailing list is active. It encourages people to participate, ask and answer questions.

## **User ML Authors**

ID: MLS\_USR\_AUTHORS\_1Y

Value: 133

Description: The total number of different identities found in the User mailing list during last year. Having many different authors is a sign of diversity and activity. It makes the support more reliable (i.e. increased presence of people) and more complete (i.e. more eyes to solve a problem).

## **Stack Overflow Views (5Y)**

ID: SO\_VIEWS\_VOL\_5Y

Value: 24691

Description: The total number of views for questions related to the project's tag on Stack Overflow during the last 5 years. Having many views on questions about the project indicates a strong interest from the community. The list of questions and their answers associated to the tag can be browsed on the Stack Overflow web site.

## **Number of jobs**

ID: CI\_JOBS

Value: 121

Description: The total number of jobs defined on the Hudson engine.



## Package Tangle index

ID: SQ\_PACKAGES\_TANGLE\_IDX

Value: 21.9 ( 5 / 5 )

Description: The Package tangle index, as defined in SonarQube.

## Number of red jobs

ID: CI\_JOBS\_RED

Value: 4

Description: The number of red (failed) jobs on the Hudson engine. Red jobs in Hudson define failed builds.

## User ML Authors

ID: MLS\_USR\_AUTHORS\_1W

Value: 6

Description: The total number of different identities found in the User mailing list during last week. Having many different authors is a sign of diversity and activity. It makes the support more reliable (i.e. increased presence of people) and more complete (i.e. more eyes to solve a problem).

## Stack Overflow Askers (5Y)

ID: SO\_ASKERS\_5Y

Value: 60

Description: The number of distinct people asking questions related to the project's tag posted on Stack Overflow during the last 5 years. Having many people ask questions about the project indicates a strong interest from the community, and a good support. The list of questions and their answers associated to the tag can be browsed on the Stack Overflow web site.

## Project authors one month

ID: PROJECT\_AUTHORS\_1M

Value:

Description: Total number of identities found as authors of commits in source code management repositories dated during the last month. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## Public API

ID: SQ\_PUBLIC\_API

Value: 17126

Description: Number of public Classes + number of public Functions + number of public Properties

## Number of functions

ID: SQ\_FUNCS

Value: 28649

Description: Number of functions. Depending on the language, a function is either a function or a method or a paragraph. For Java, constructors are considered as methods and accessors are considered as methods if the sonar.squid.analyse.property.accessors property is set to false. For Cobol, it is the number of paragraphs.

## SCM Still Open Pull requests one year

ID: SCM\_PRS\_OPENED\_STILL\_1Y

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) that have been opened more than one year ago and are still in the opened state in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## User ML Posts

ID: MLS\_USR\_POSTS

Value: 2609

Description: The total number of posts found in the User mailing list. Having many posts shows the mailing list is active. It encourages people to participate, ask and answer questions.

## User ML Threads

ID: MLS\_USR\_THREADS\_1W

Value: 7

Description: The total number of threads (one question followed by zero or more answers) found in the User mailing list during last week. Having many threads shows the mailing list is active. It encourages people to participate, ask and answer questions.

## SCM Open Pull requests one year

ID: SCM\_PRS\_OPENED\_1Y

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) that have been opened within the last year in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## User ML Threads

ID: MLS\_USR\_THREADS\_1M

Value: 33 ( 5 / 5 )

Description: The total number of threads (one question followed by zero or more answers) found in the User mailing list during last month. Having many threads shows the mailing list is active. It encourages people to participate, ask and answer questions.

## Number of yellow jobs

ID: CI\_JOBS\_YELLOW

Value: 4

Description: The number of yellow (unstable) jobs on the Hudson engine. Yellow jobs in Hudson define unstable builds. According to Hudson's documentation, a build is unstable if it was built successfully and one or more publishers report it unstable. For example if the JUnit publisher is configured and a test fails then the build will be marked unstable.

## Number of stars

ID: PROJECT\_STARS

Value:

Description: The number of times people have starred this project. Users use stars to show their interest for a project, and more stars usually mean a greater visibility and interest.

## ITS Open issues (%)

ID: ITS\_OPEN\_PERCENT

Value: 21

Description: Percentage of open issues compared to the overall number of issues registered in the system.

## SCM authors one year

ID: SCM\_AUTHORS\_1Y

Value: 15

Description: Total number of identities found as authors of commits in source code management repositories dated during the last year. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## **File complexity**

ID: SQ\_CPX\_FILE\_IDX

Value: 18 ( 2 / 5 )

Description: Average complexity by file.

## **SCM authors**

ID: SCM\_AUTHORS

Value: 121

Description: Total number of identities found as authors of commits in source code management repository. Source code management repositories are those considered as such in the project documentation. Commits in all branches are considered. Date used for each commit is 'author date' (when there is a difference between author date and committer date). An identity is considered as author if it appears as such in the commit record (for systems logging several identities related to the commit, authoring identity will be considered).

## **Stack Overflow Votes (5Y)**

ID: SO\_VOTES\_VOL\_5Y

Value: 67

Description: The total number of votes on questions related to the project's tag on Stack Overflow during the last 5 years. Having many votes on questions about the project indicates a strong interest from the community. The list of questions and their answers associated to the tag can be browsed on the Stack Overflow web site.

## **Number of failed jobs one week**

ID: CI\_JOBS\_FAILED\_1W

Value: 4 ( 3 / 5 )

Description: The number of jobs that failed during last week on the Hudson engine.

## **Project Commits one year**

ID: PROJECT\_COMMITS\_1Y

Value:

Description: Total number of commits in source code management repositories dated during the last year. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## **SCM Staled Open Pull requests one month**

ID: SCM\_PRS\_OPENED\_STALED\_1M

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) that are in the opened state and have not been updated since one month. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## **SCM Open Pull requests one week**

ID: SCM\_PRS\_OPENED\_1W

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) that have been opened within the last week in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.

## **Project Commits**

ID: PROJECT\_COMMITS

Value:

Description: Total number of commits in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date).

## **CI information**

ID: PROJECT\_CI\_INFO

Value:

Description: Is the continuous integration info correctly filled in the PMI records? The project management infrastructure file holds information about the location of CI services. This test checks the number of ci-related entries defined in the PMI.

## **Project authors one week**

ID: PROJECT\_AUTHORS\_1W

Value:

Description: Total number of identities found as authors of commits in source code management repositories dated during the last week. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.

## **Last activity**

ID: PROJECT\_LAST\_ACTIVITY\_AT

Value:

Description: The date of last activity for this project. This includes any type of action: changes on issues, git commits or pushes, merge requests, comments... A project with a old last activity timestamp shows the project is dead (or in agony at least).

## **Test coverage**

ID: SQ\_COVERAGE

Value:

Description: Overall test coverage.

## **Duplicated lines (%)**

ID: SQ\_DUPLICATED\_LINES\_DENSITY

Value: 6.3 ( 5 / 5 )

Description: Density of duplication = Duplicated lines / Lines \* 100.

## **SCM Merged Pull requests**

ID: SCM\_PRS\_MERGED

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) in the 'merged' state in source code management repositories.

## **SCM Closed Pull requests**

ID: SCM\_PRS\_CLOSED

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) in the closed state in source code management repositories.

## **Stack Overflow Questions (5Y)**

ID: SO\_QUESTIONS\_VOL\_5Y

Value: 70

Description: The number of questions related to the project's tag posted on Stack Overflow during the last 5 years. Having many questions posted about the project indicates a strong interest from the community. The list of questions associated to the tag can be browsed on the Stack Overflow web site.

## **Number of minor issues**

ID: SQ\_VIOLATIONS\_MINOR

Value: 3685

Description: The total number of issues (violations) found by SonarQube with a severity equal to MINOR.

## **Project committers**

ID: PROJECT\_COMMITTERS

Value:

Description: Total number of identities found as committers of commits in source code management repository. Source code management repositories are those considered as such in the project documentation. Commits in all branches are considered. Date used for each commit is 'committer date' (when there is a difference between author date and committer date). An identity is considered as committer if it appears as such in the commit record.

## **Sqale Debt ratio**

ID: SQ\_SQALE\_DEBT\_RATIO

Value: 3.9 ( 5 / 5 )

Description: The Technical Debt Ratio, as defined in Sqale.

## **ITS issues created last year**

ID: ITS\_CREATED\_1Y

Value: 524

Description: Number of issues created during last year. If today is 2017-02-01 then the range is from 2016-02-01 to 2017-02-01.

## **SCM committers**

ID: SCM\_COMMITTERS

Value: 98

Description: Total number of identities found as committers of commits in source code management repository. Source code management repositories are those considered as such in the project documentation. Commits in all branches are considered. Date used for each commit is 'committer date' (when there is a difference between author date and committer date). An identity is considered as committer if it appears as such in the commit record.

## **User ML Authors**

ID: MLS\_USR\_AUTHORS

Value: 1254

Description: The total number of different identities found in the User mailing list. Having many different authors is a sign of diversity and activity. It makes the support more reliable (i.e. increased presence of people) and more complete (i.e. more eyes to solve a problem).

### **ITS authors last year**

ID: ITS\_AUTHORS\_1Y

Value: 44

Description: Number of authors who created issues during last year. If today is 2017-02-01 then the range is from 2016-02-01 to 2017-02-01.

### **ITS issues created last month**

ID: ITS\_CREATED\_1M

Value: 31

Description: Number of issues created during last month. If today is 2017-02-01 then the range is from 2017-01-01 to 2017-02-01.

### **Maintainability rating**

ID: SQ\_SQALE\_RATING

Value: 1

Description: Rating given to your project related to the value of your Technical Debt Ratio. The default Maintainability Rating grid is: A=0-0.05, B=0.06-0.1, C=0.11-0.20, D=0.21-0.5, E=0.51-1. The Maintainability Rating scale can be alternately stated by saying that if the outstanding remediation cost depends on the time that has already gone into the application: A  $\leq 5\%$ , B between 6 to 10%, C between 11 to 20%, D between 21 to 50%, and anything over 50% is an E.

### **ITS Authors**

ID: ITS\_AUTHORS

Value: 493

Description: Number of different authors who created issues during the lifetime of the project. A high number of authors shows diversity and improves the bus factor of the project.

### **ITS issues updated last month**

ID: ITS\_UPDATED\_1M

Value: 31 ( 4 / 5 )

Description: Number of issues updated during last month. If today is 2017-02-01 then the range is from 2017-01-01 to 2017-02-01.



## **User ML Threads**

ID: MLS\_USR\_THREADS\_1Y

Value: 191

Description: The total number of threads (one question followed by zero or more answers) found in the User mailing list during last year. Having many threads shows the mailing list is active. It encourages people to participate, ask and answer questions.

## **ITS Late issues**

ID: ITS\_LATE

Value:

Description: Number of issues with a past due date. It is considered good practice to keep this number low. Either fix it or maintain its due date.

## **ITS Pending issues**

ID: ITS\_OPEN\_UNASSIGNED

Value: 0

Description: Number of issues in state open with no assignee (i.e. pending). It is considered to be good practice to keep this number low. In an active project, people would either work on the bug (i.e. assign it) or triage it (pass it to some other state or assigning it).

## **Project committers one month**

ID: PROJECT\_COMMITTERS\_1M

Value:

Description: Total number of identities found as committers of commits in source code management repositories dated during the last month. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'committer date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## **Project authors one year**

ID: PROJECT\_AUTHORS\_1Y

Value:

Description: Total number of identities found as authors of commits in source code management repositories dated during the last year. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## User ML Posts

ID: MLS\_USR\_POSTS\_1W

Value: 14

Description: The total number of posts found in the User mailing list during last week. Having many posts shows the mailing list is active. It encourages people to participate, ask and answer questions.

## Open issues

ID: PROJECT\_ISSUES\_OPEN

Value:

Description: The number of issues opened at the time of analysis on the GitLab project. This information is retrieved from GitLab itself, and may differ from numbers gathered from the actual issue tracking system used.

## ITS authors last month

ID: ITS\_AUTHORS\_1M

Value: 9 ( 3 / 5 )

Description: Number of authors who created issues during last month. If today is 2017-02-01 then the range is from 2017-01-01 to 2017-02-01.

## ITS issues updated last year

ID: ITS\_UPDATED\_1Y

Value: 524

Description: Number of issues updated during last year. If today is 2017-02-01 then the range is from 2016-02-01 to 2017-02-01.

## Number of forks

ID: PROJECT\_FORKS

Value:

Description: The number of forks for this project. More forks usually mean a greater activity.

## Ratio of green jobs

ID: CI\_JOBS\_GREEN\_RATIO

Value: 92 ( 4 / 5 )

Description: The number of green (successful) jobs on the Hudson engine, divided by the total number of jobs. Green (or blue) jobs in Hudson define successful builds.

## Project committers one week

ID: PROJECT\_COMMITTERS\_1W

Value:

Description: Total number of identities found as committers of commits in source code management repositories dated during the last week. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'committer date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.

## Branch coverage

ID: SQ\_COVERAGE\_BRANCH

Value:

Description: Branch test coverage.

## Access information

ID: PROJECT\_ACCESS\_INFO

Value: 3 ( 5 / 5 )

Description: Is the access info (downloads, update sites..) correctly filled in the PMI records? The project management infrastructure file holds information about how to access binaries of the project. This test checks the number of access-related entries defined in the PMI: download\_url, downloads, update\_sites.

## Project authors

ID: PROJECT\_AUTHORS

Value:

Description: Total number of identities found as authors of commits in source code management repository. Source code management repositories are those considered as such in the project documentation. Commits in all branches are considered. Date used for each commit is 'author date' (when there is a difference between author date and committer date). An identity is considered as author if it appears as such in the commit record (for systems logging several identities related to the commit, authoring identity will be considered).

## Comment lines density

ID: SQ\_COMR

Value: 27.6

Description: Density of comment lines = Comment lines / (Lines of code + Comment lines) \* 100. With such a formula, 50% means that the number of lines of code equals the number of comment lines and 100% means that the file only contains comment lines

## Number of comment lines

ID: SQ\_COMMENT\_LINES

Value: 122797

Description: Number of lines containing either comment or commented-out code. Non-significant comment lines (empty comment lines, comment lines containing only special characters, etc.) do not increase the number of comment lines. For Java, file headers are not counted as comment lines (as they usually define the license). Lines containing the following instructions are counted both as comments and lines of code: AUTHOR, INSTALLATION, DATE-COMPILED, DATE-WRITTEN, SECURITY.

## Number of major issues

ID: SQ\_VIOLATIONS\_MAJOR

Value: 13496

Description: The total number of issues (violations) found by SonarQube with a severity equal to MAJOR.

## Number of lines of code

ID: SQ\_NCLOC

Value: 321906

Description: Number of physical lines that contain at least one character which is neither a whitespace or a tabulation or part of a comment. For Cobol, generated lines of code and pre-processing instructions (SKIP1, SKIP2, SKIP3, COPY, EJECT, REPLACE) are not counted as lines of code.

## Number of releases

ID: PROJECT\_REL\_VOL

Value: 32 ( 5 / 5 )

Description: The number of releases recorded in the PMI. Milestones are retrieved from the PMI file and are counted whatever their target release is. Milestones are useful to assess the maturity of the release and improves predictability of the project's output, in terms of quality and time.

## SCM Changed Lines one year

ID: SCM\_MOD\_LINES\_1Y

Value: 860817

Description: Total number of changed lines (added, removed, changed) in source code management repositories dated during the last year. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one year period starting the day before the data retrieval (example: if retrieval is on Feb 3rd 2016, period is from Feb 3rd 2015 to Feb 3rd 2016, both included).

## SCM Pull requests

ID: SCM\_PRS

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) in source code management repositories. Source code management repositories are those considered as such in the project documentation.

## SCM authors one month

ID: SCM\_AUTHORS\_1M

Value: 5 ( 4 / 5 )

Description: Total number of identities found as authors of commits in source code management repositories dated during the last month. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## SCM Still Open Pull requests one week

ID: SCM\_PRS\_OPENED\_STILL\_1W

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) that have been opened more than one week ago and are still in the opened state in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.

## ITS issues updated last week

ID: ITS\_UPDATED\_1W

Value: 10

Description: Number of issues updated during last week. If today is Wed. 2017-02-01 then the range is from Wed. 2017-01-25 to Wed. 2017-02-01.

## Commented code

ID: SQ\_COM\_CODE

Value:

Description: Commented lines of code See more information about commented code on SonarQube doc web site. There is a well-documented debate on Stack Overflow as well.

## **Project Commits one month**

ID: PROJECT\_COMMITS\_1M

Value:

Description: Total number of commits in source code management repositories dated during the last month. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## **SCM Commits one week**

ID: SCM\_COMMITS\_1W

Value: 9

Description: Total number of commits in source code management repositories dated during the last week. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.

## **SCM Changed Lines one month**

ID: SCM\_MOD\_LINES\_1M

Value: 14831 ( 5 / 5 )

Description: Total number of changed lines (added, removed, changed) in source code management repositories dated during the last month. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## **Stack Overflow Answer rate (5Y)**

ID: SO\_ANSWER\_RATE\_5Y

Value: 0.87

Description: The average number of answers per questions related to the project's tag on Stack Overflow during the last 5 years. Having many answers posted about the project indicates a strong interest from the community, and a good support. The list of questions and their answers associated to the tag can be browsed on the Stack Overflow web site.

## **SCM Open Pull requests**

ID: SCM\_PRS\_OPENED

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) in the opened state in source code management repositories.

## **Number of files**

ID: SQ\_FILES

Value: 4173

Description: The total number of files analysed.

## **ITS Total issues**

ID: ITS\_ISSUES\_ALL

Value: 11875

Description: Number of issues registered in the database, whatever their state is.

## **SCM Commits**

ID: SCM\_COMMITS

Value: 14845

Description: Total number of commits in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date).

## **SCM committers one week**

ID: SCM\_COMMITTERS\_1W

Value: 3

Description: Total number of identities found as committers of commits in source code management repositories dated during the last week. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'committer date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.

## **Number of info issues**

ID: SQ\_VIOLATIONS\_INFO

Value: 1812

Description: The total number of issues (violations) found by SonarQube with a severity equal to INFO.

## **User ML Posts**

ID: MLS\_USR\_POSTS\_1Y

Value: 690

Description: The total number of posts found in the User mailing list during last year. Having many posts shows the mailing list is active. It encourages people to participate, ask and answer questions.

## **Line coverage**

ID: SQ\_COVERAGE\_LINE

Value:

Description: Line test coverage.

## **SCM Still Open Pull requests one month**

ID: SCM\_PRS\_OPENED\_STILL\_1M

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) that have been opened more than one month ago and are still in the opened state in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## **Number of releases**

ID: PMI\_REL\_VOL

Value:

Description: The number of releases recorded in the PMI. Milestones are retrieved from the PMI file and are counted whatever their target release is. Milestones are useful to assess the maturity of the release and improves predictability of the project's output, in terms of quality and time.

## **Stack Overflow Answers (5Y)**

ID: SO\_ANSWERS\_VOL\_5Y

Value: 61

Description: The number of answers to questions related to the project's tag posted on Stack Overflow during the last 5 years. Having many answers posted about the project indicates a strong interest from the community, and a good support. The list of questions and their answers associated to the tag can be browsed on the Stack Overflow web site.

## **Project Commits one week**

ID: PROJECT\_COMMITS\_1W

Value:

Description: Total number of commits in source code management repositories dated during the last week. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one week period starting the day before the data retrieval.



## SCM Open Pull requests one month

ID: SCM\_PRS\_OPENED\_1M

Value:

Description: Total number of Pull Requests (PRs) or Merge Requests (MRs) that have been opened within the last month in source code management repositories. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'author date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## ITS authors last week

ID: ITS\_AUTHORS\_1W

Value: 5

Description: Number of authors who created issues during last week. If today is Wed. 2017-02-01 then the range is from Wed. 2017-01-25 to Wed. 2017-02-01.

## Total complexity

ID: SQ\_CPX

Value: 75102

Description: It is the complexity calculated based on the number of paths through the code. Whenever the control flow of a function splits, the complexity counter gets incremented by one. Each function has a minimum complexity of 1. This calculation varies slightly by language because keywords and functionalities do. For more information on line counting for each language, see <https://docs.sonarqube.org/display/SONAR/Metrics+-+Complexity>.

## SCM committers one month

ID: SCM\_COMMITTERS\_1M

Value: 5

Description: Total number of identities found as committers of commits in source code management repositories dated during the last month. Source code management repositories are those considered as such in the project documentation. Date used for each commit is 'committer date' (when there is a difference between author date and committer date). Time range is measured as a one month period starting the day before the data retrieval (example: if retrieval is on Feb 3rd, period is from Jan 3rd to Feb 2nd, both included).

## User ML Authors

ID: MLS\_USR\_AUTHORS\_1M

Value: 24 ( 5 / 5 )

Description: The total number of different identities found in the User mailing list during last month. Having many different authors is a sign of diversity and activity. It makes the support more reliable (i.e. increased presence of people) and more complete (i.e. more eyes to solve a problem).

## Number of statements

ID: SQ\_STATEMENTS

Value: 135946

Description: Number of statements. For Java, it is the number of statements as defined in the Java Language Specification but without block definitions. Statements counter gets incremented by one each time a following keyword is encountered: if, else, while, do, for, switch, break, continue, return, throw, synchronized, catch, finally.. Statements counter is not incremented by a class, method, field, annotation definition, package declaration and import declaration. For Cobol, a statement is one of move, if, accept, add, alter, call, cancel, close, compute, continue, delete, display, divide, entry, evaluate, exitProgram, goback, goto, initialize, inspect, merge, multiply, open, perform, read, release, return, rewrite, search, set, sort, start, stop, string, subtract, unstring, write, exec, ibmXmlParse, ibmXmlGenerate, readyReset, mfCommit, mfRollback.

## Number of blocker issues

ID: SQ\_VIOLATIONS\_BLOCKER

Value: 2 ( 4 / 5 )

Description: The total number of issues (violations) found by SonarQube with a severity equal to BLOCKER.

## Attributes

Mnemo	Value
QM_ACTIVITY	4.20
QM_DIVERSITY	4.00
QM_DOC	4.50
QM_ECOSYSTEM	4.30
QM_MAINTAINABILITY	5.00
QM_PROCESS	4.30
QM_PRODUCT	4.30
QM_QUALITY	4.30
QM_RELIABILITY	3.70
QM_REL_ENG	4.50
QM_SCM	4.00
QM_SUPPORT	4.70

## Activity

ID: QM\_ACTIVITY

Value: 4.2 / 5

Description: The activity of the project's ecosystem, as measured on the mailing lists and configuration management system. An active project will provide a lot of information on the mailing lists, so when an user encounters an issue she will quickly find the information she needs, and has more chances to get answers if she asks. Fixes and improvements are added regularly.

## Diversity

ID: QM\_DIVERSITY

Value: 4 / 5

Description: The diversity of the project's ecosystem, as measured on the mailing lists and configuration management system. If many different actors from different companies are involved in the project, then it

improves its sustainability (by eliminating a single point of failure) and adaptability to different situations. Having developers and users with different contexts and perspectives on the project helps widening its scope and provide a more generic support.

## **Documentation**

ID: QM\_DOC

Value: 4.5 / 5

Description: The maturity of code. Good code is vital for maintenance and evolution. It will encourage people to contribute, lower the number of bugs, and make a better product for the end-user as well as for the maintainers.

## **Ecosystem**

ID: QM\_ECOSYSTEM

Value: 4.3 / 5

Description: The sustainability of the ecosystem evolving around the project. Sustainability is a key point for long term support. If there is a lot of activity, if people can get fast and complete answers, if many people from different companies contribute to the project, then it will have more chance to still be there in a few years, and to continue providing fixes and improvements. Ecosystem requirements have been discussed on the mailing list and during meetings, and have been further described on the Polarsys wiki.

## **Maintainability**

ID: QM\_MAINTAINABILITY

Value: 5 / 5

Description: The Maintainability of the codebase, as defined in ISO-9126.

## **Process**

ID: QM\_PROCESS

Value: 4.3 / 5

Description: The maturity of the process used to run the project. A sound process helps people to do things right and ease collaborative work. If the process is documented, has predictable output, helps enforcing good development practices, etc. then new comers will easily find the information to collaborate, test or change code, or participate in the community. A good process also helps producing a good product [[Ing2003](/documentation/references.html#Ing2003)] – although it is agreed that the process is not enough by itself. Process requirements have been discussed on the mailing list and during meetings, and have been further described on the Polarsys wiki. Some may also recognise CMMi Key Process Areas among the attributes.

## **Product**

ID: QM\_PRODUCT

Value: 4.3 / 5

Description: The maturity of the product itself, from the code perspective. Considering the vast amount and diversity of the projects under the Eclipse umbrella, there must be no single definition of quality to fit them all. However, Eclipse has some recommended practices and concerns about product quality. Projects are then expected to extend this foundation. Major concerns identified for Eclipse products quality are linked to the development context of the foundation (open source, very large code base and thousands of contributor worldwide), and its architecture (modular stacks of components). It must be highlighted that product quality is not clearly defined on the public wiki, neither for its definition nor for how it may be assessed. Furthermore, almost all product-related rules (with a few exceptions, like for packages naming) are optional guidelines. . Ecosystem requirements have been discussed on the mailing list and during meetings, and have been further described on the Polarsys wiki.

## **Eclipse Maturity**

ID: QM\_QUALITY

Value: 4.3 / 5

Description: The overall Maturity of the project. In the context of embedded software, Maturity is usually associated with some kind of reliability (most bugs have been already found) and functionality of code, sustainability of the project (will it still deliver fixes and improvements in a few years), and process predictability. Maturity in the PolarSys context has been further described on the wiki, and is actually precisely defined by the decomposition of this quality model.

## **Reliability**

ID: QM\_RELIABILITY

Value: 3.7 / 5

Description: The Reliability of code, as defined in ISO-9126.

## **Build and Release Management**

ID: QM\_REL\_ENG

Value: 4.5 / 5

Description: Does the project apply best practices regarding Build and Release management?

## **Configuration Management**

ID: QM\_SCM

Value: 4 / 5

Description: The maturity of the project regarding access and usage of the configuration management system. Configuration management is an essential part of the collaboration in the project. Access to the source should be documented and facilitated for new comers to easily come in.

## **Support**

ID: QM\_SUPPORT

Value: 4.7 / 5

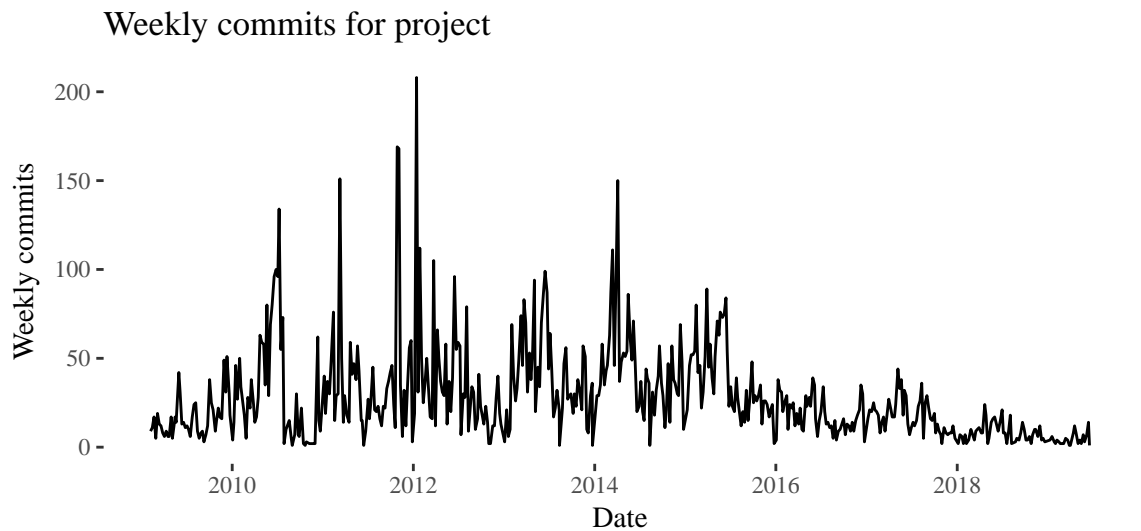
Description: The amount of knowledge provided when someone asks for support. Having many answers on a single question helps better understand how the product works in different conditions, and also provides help for people looking for a similar information later on, since mailing lists are archived and public.

## Git analysis

The repository contains a total of 14845 commits made by 121 authors. The first commit was made on the 2009-02-06 and the last analysed commits was made on 2019-06-18.

During the last month, there has been 32 commits made by 5 authors.

## Weekly commits



## Weekly authors

