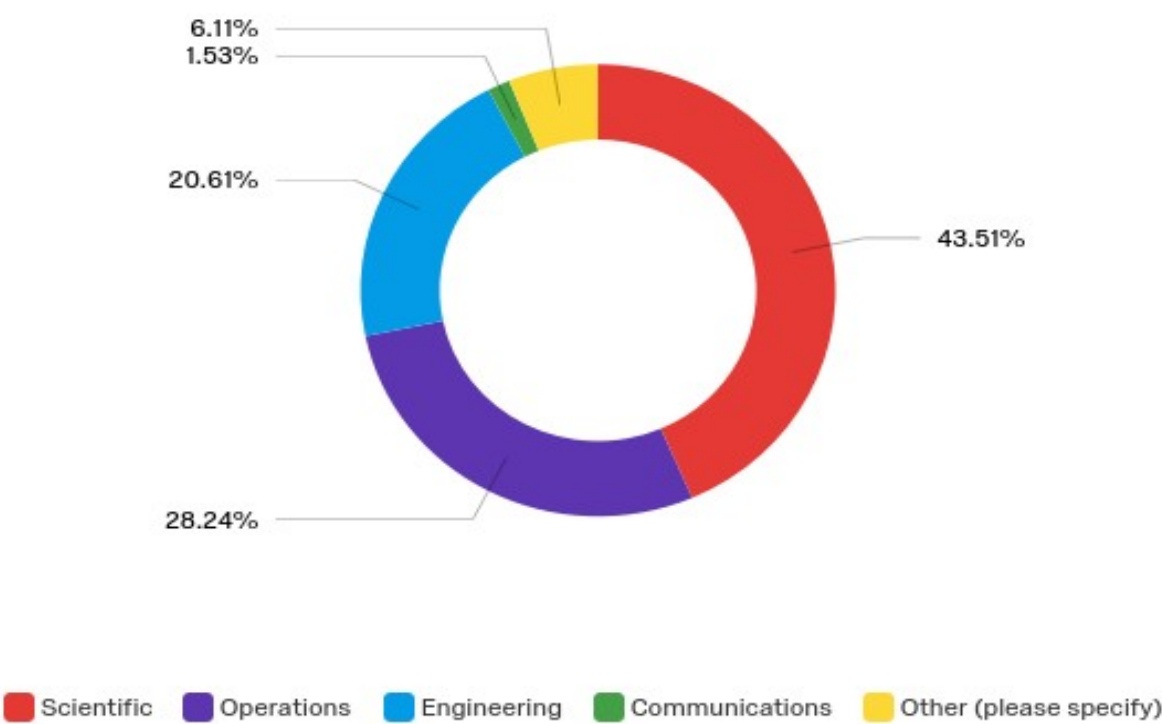


First Results

SLAC Software Development Tools

February 10th 2017, 4:01 pm MST

Q11 - In which capacities do you write software?



Q11_5_TEXT - Other (please specify)

Other (please specify)
Professional Staff
KIPAC/Campus
KIPAC/Campus
KIPAC/Campus
Computer Center
Controls
EPICS control for LCLS and Photon Exp Halls
Computing infrastructure

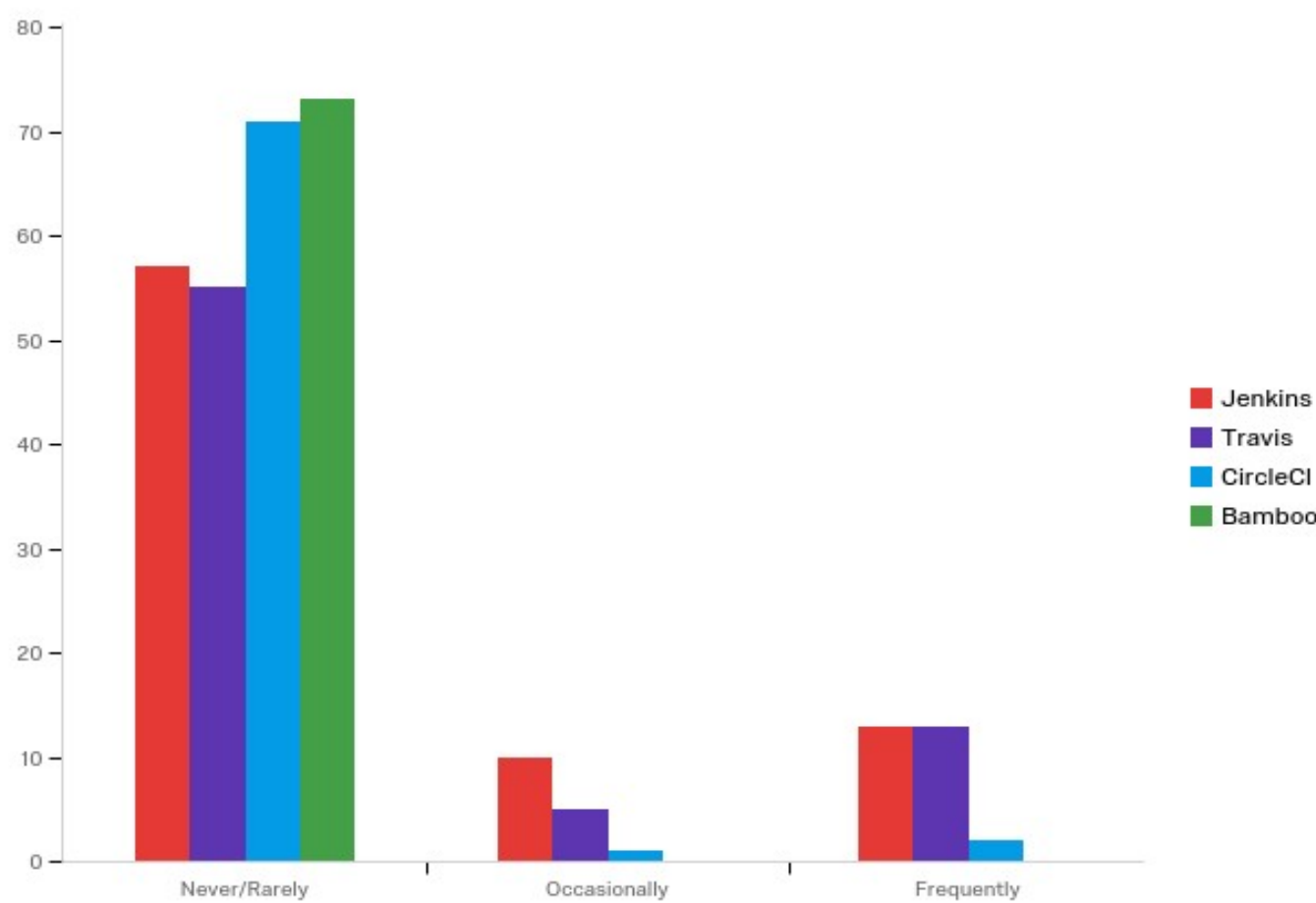
Q12 - What programming languages do you use?

Programming Languages used at SLAC

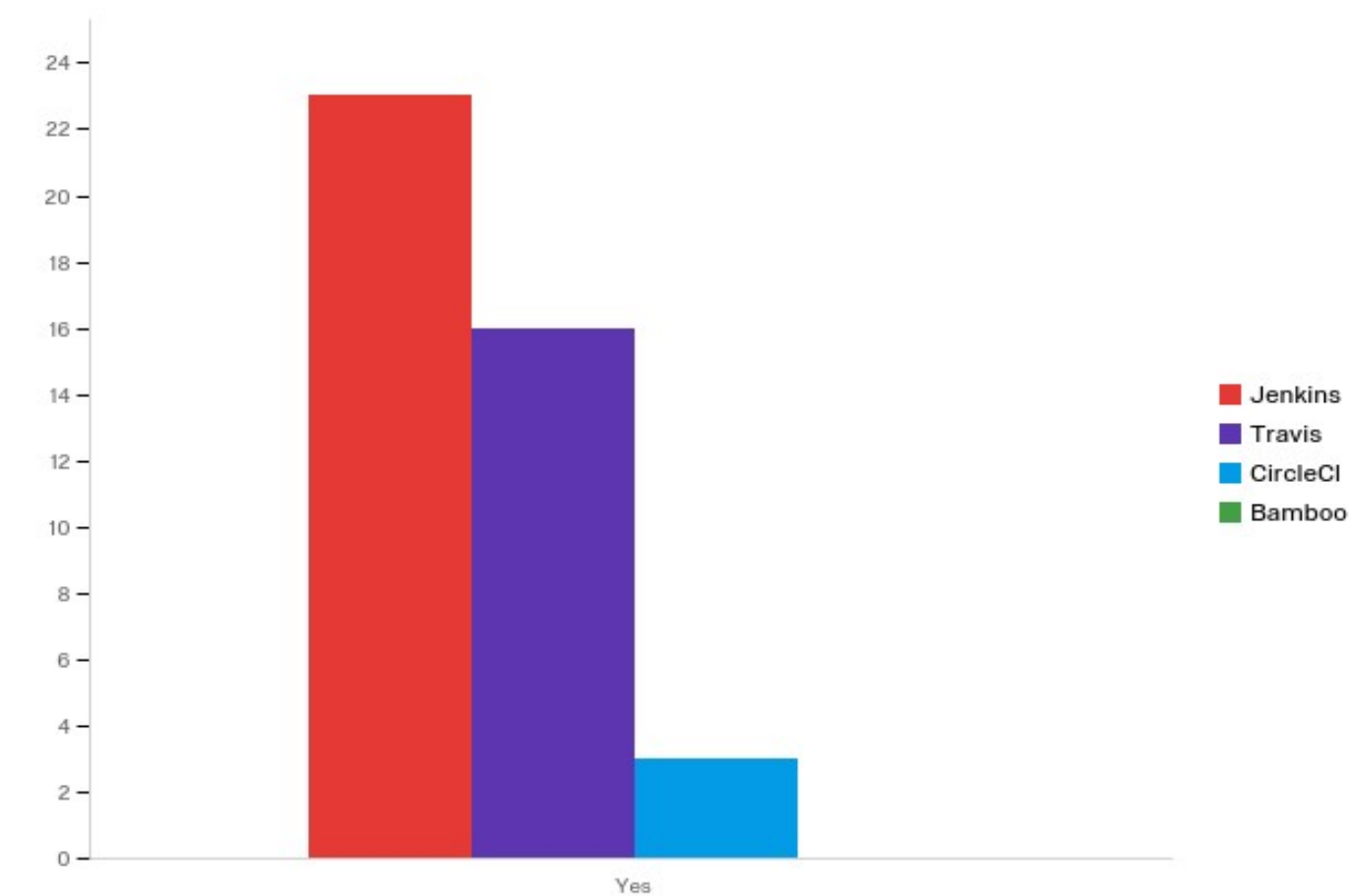


Q1#1 - How often have you used it?

How often have you use these CI tools?

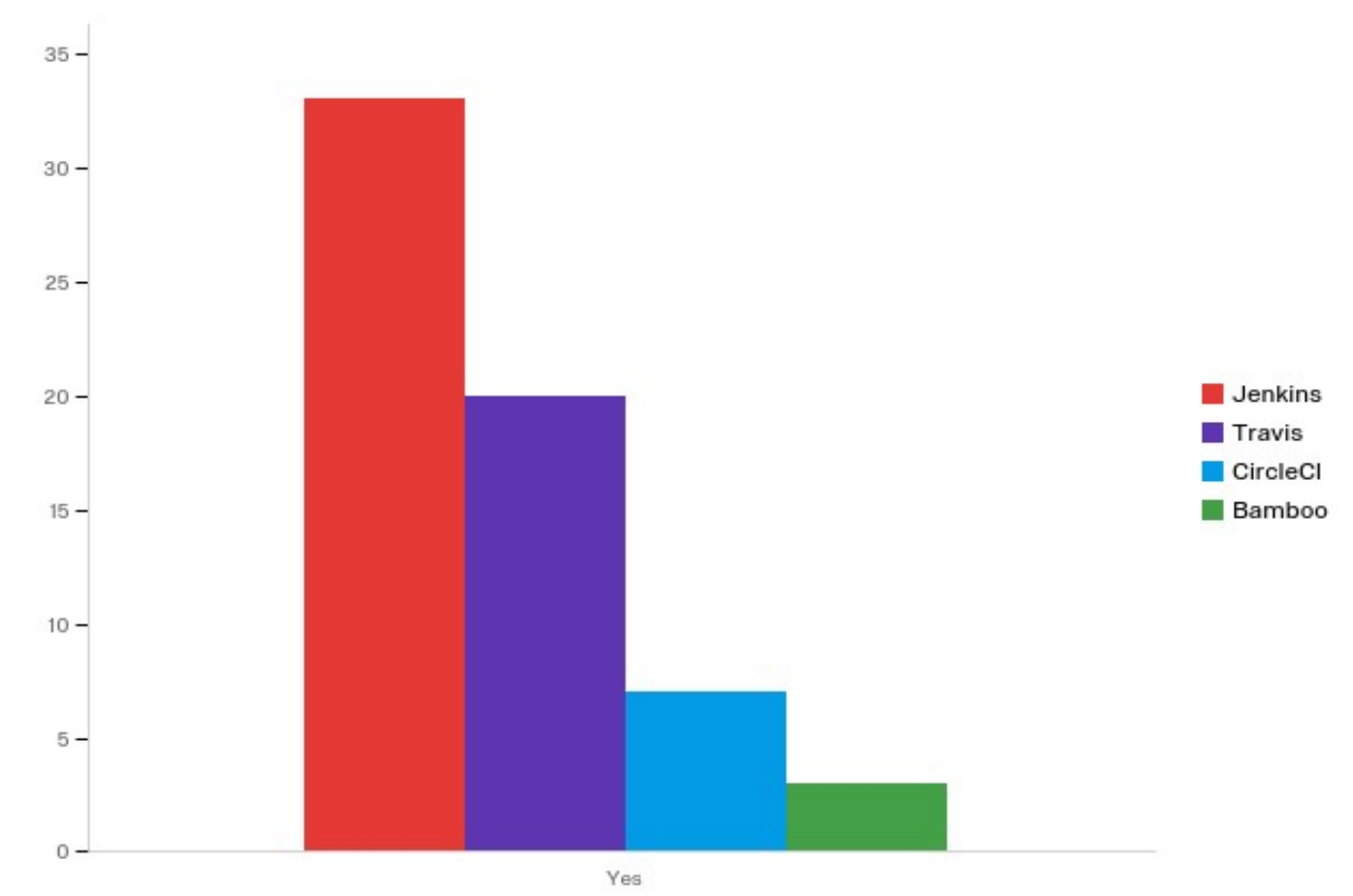


Q1#2 - Do you currently use it?



#	Question	Yes		Total
1	Jenkins	100.00%	23	23
2	Travis	100.00%	16	16
3	CircleCI	100.00%	3	3
4	Bamboo	0.00%	0	0

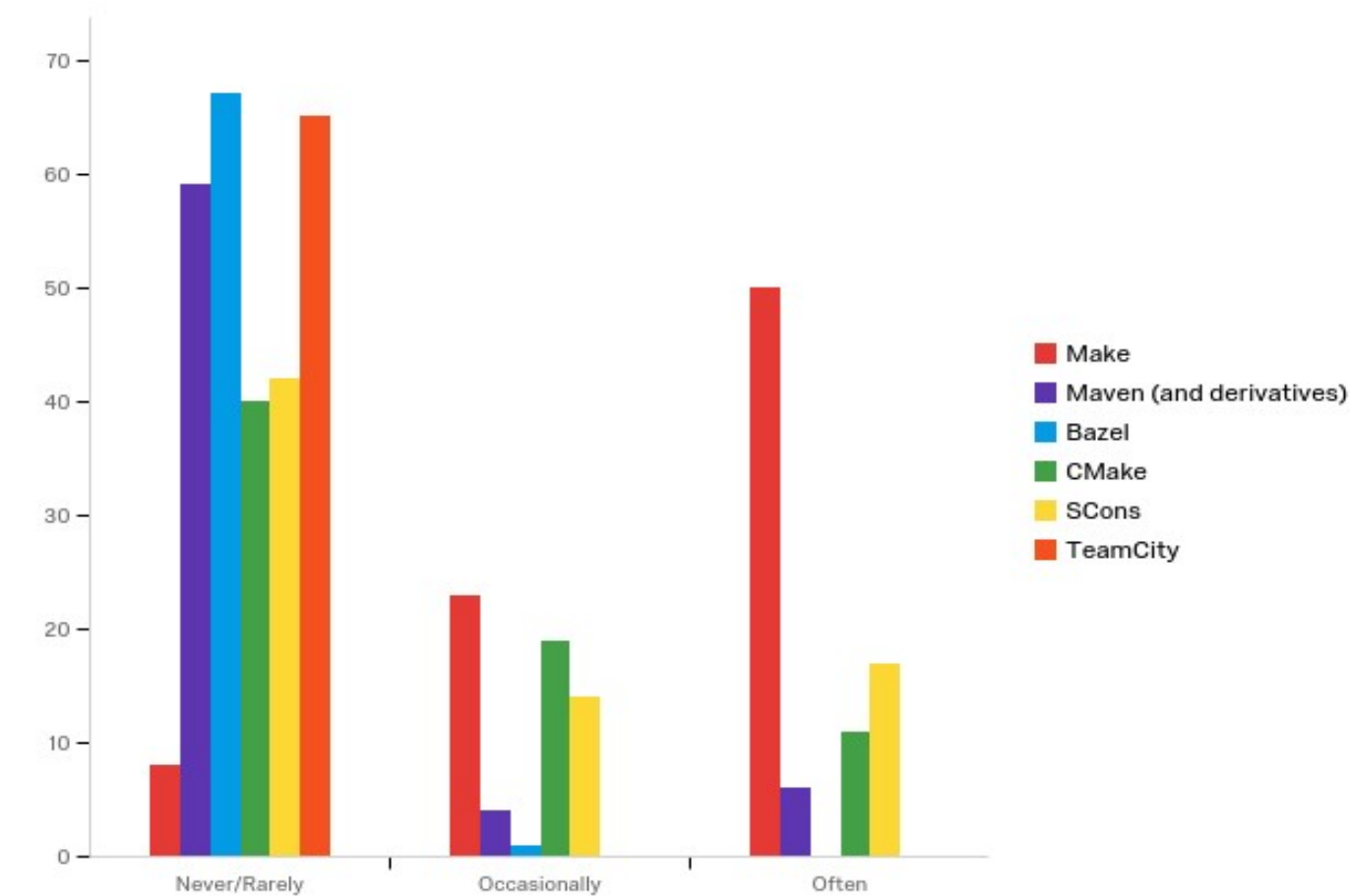
Q1#3 - Would you like to continue/start using it?



#	Question	Yes		Total
1	Jenkins	100.00%	33	33
2	Travis	100.00%	20	20
3	CircleCI	100.00%	7	7
4	Bamboo	100.00%	3	3

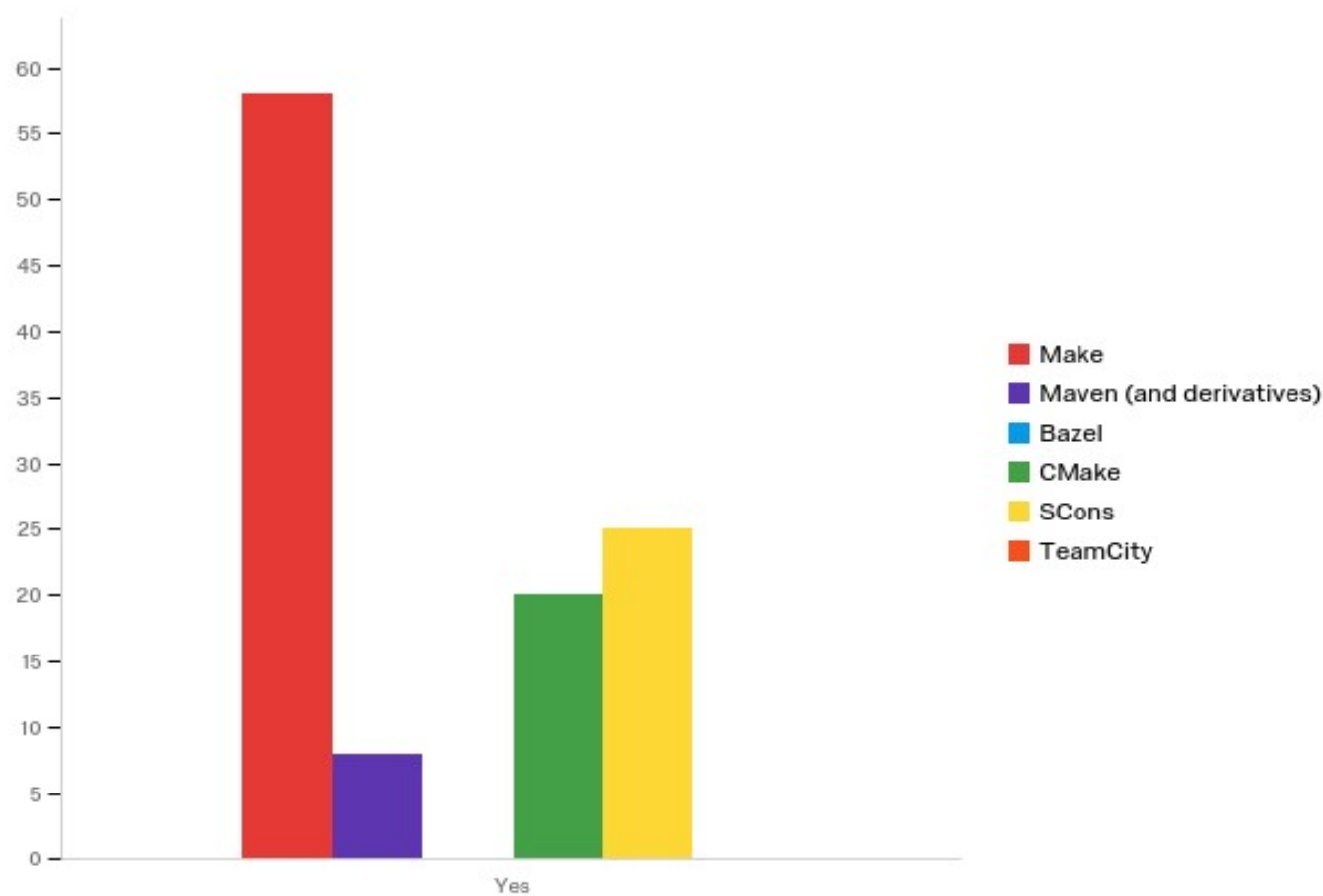
Q3#1 - How often have you used it?

How often have you used these build tools?



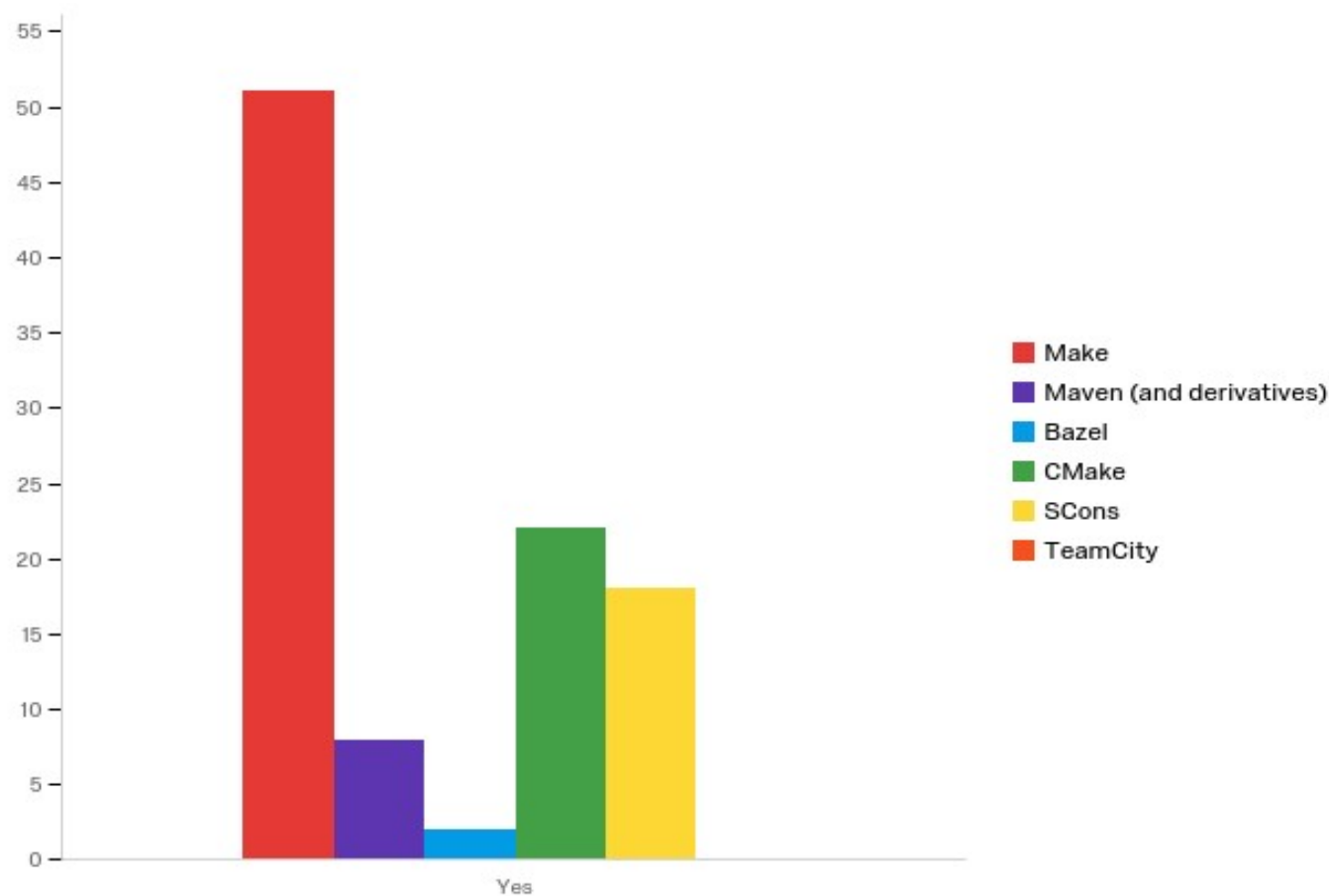
#	Question	Never/Rarely		Occasionally		Often		Total
1	Make	9.88%	8	28.40%	23	61.73%	50	81
2	Maven (and derivatives)	85.51%	59	5.80%	4	8.70%	6	69
3	Bazel	98.53%	67	1.47%	1	0.00%	0	68
4	CMake	57.14%	40	27.14%	19	15.71%	11	70
5	SCons	57.53%	42	19.18%	14	23.29%	17	73
6	TeamCity	100.00%	65	0.00%	0	0.00%	0	65

Q3#2 - Do you currently use it?



#	Question	Yes		Total
1	Make	100.00%	58	58
2	Maven (and derivativ es)	100.00%	8	8
3	Bazel	0.00%	0	0
4	CMake	100.00%	20	20
5	SCons	100.00%	25	25
6	TeamCity	0.00%	0	0

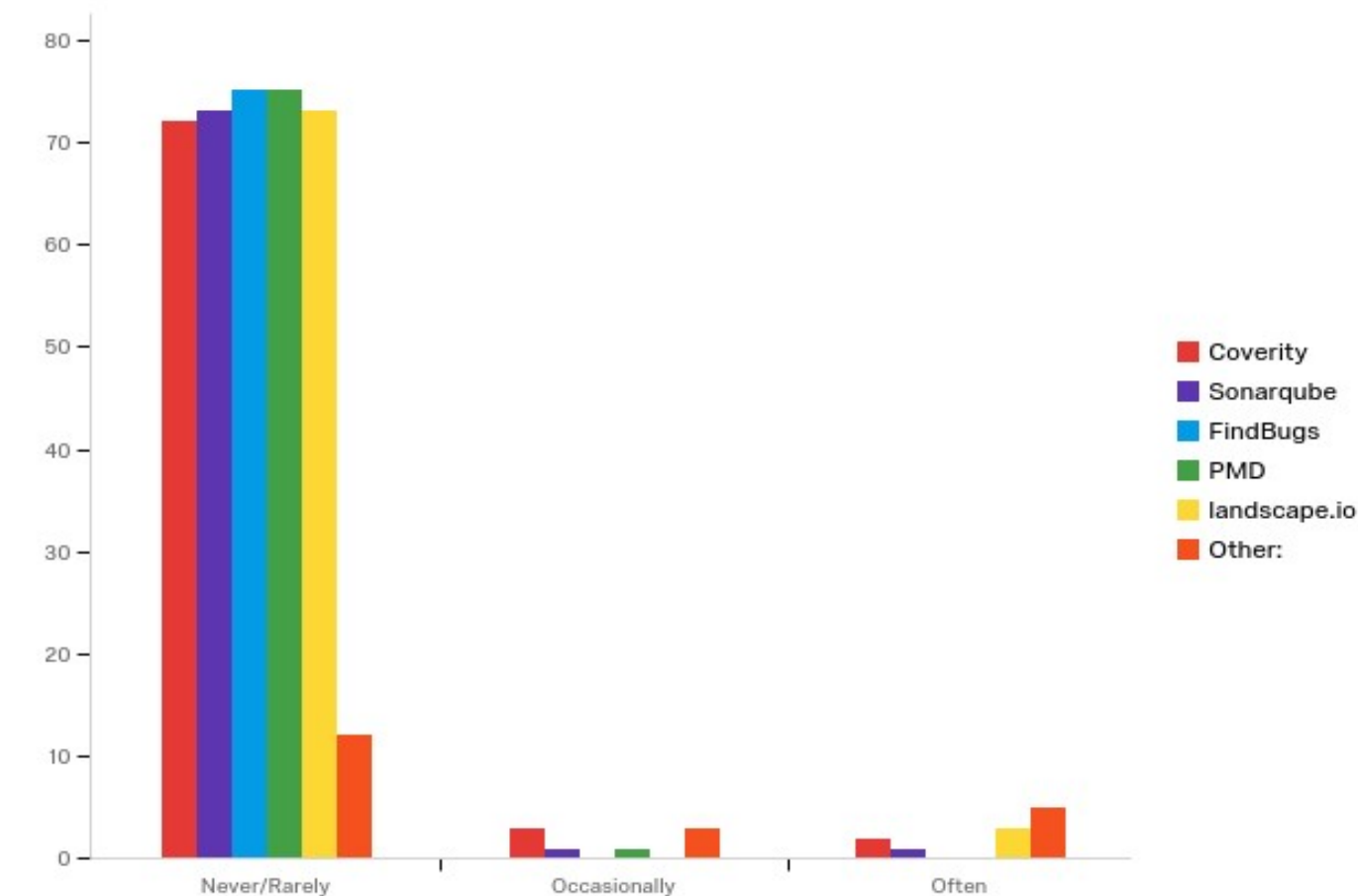
Q3#3 - Would you like to continue/start using it?



#	Question	Yes		Total
1	Make	100.00%	51	51
2	Maven (and derivatives)	100.00%	8	8
3	Bazel	100.00%	2	2
4	CMake	100.00%	22	22
5	SCons	100.00%	18	18
6	TeamCity	0.00%	0	0

Q4#1 - How often have you used it?

How often have use used these code quality tools?



#	Question	Never/Rarely		Occasionally		Often		Total
1	Coverity	93.51%	72	3.90%	3	2.60%	2	77
2	Sonarqube	97.33%	73	1.33%	1	1.33%	1	75
3	FindBugs	100.00%	75	0.00%	0	0.00%	0	75
4	PMD	98.68%	75	1.32%	1	0.00%	0	76
5	landscape.io	96.05%	73	0.00%	0	3.95%	3	76
6	Other:	60.00%	12	15.00%	3	25.00%	5	20

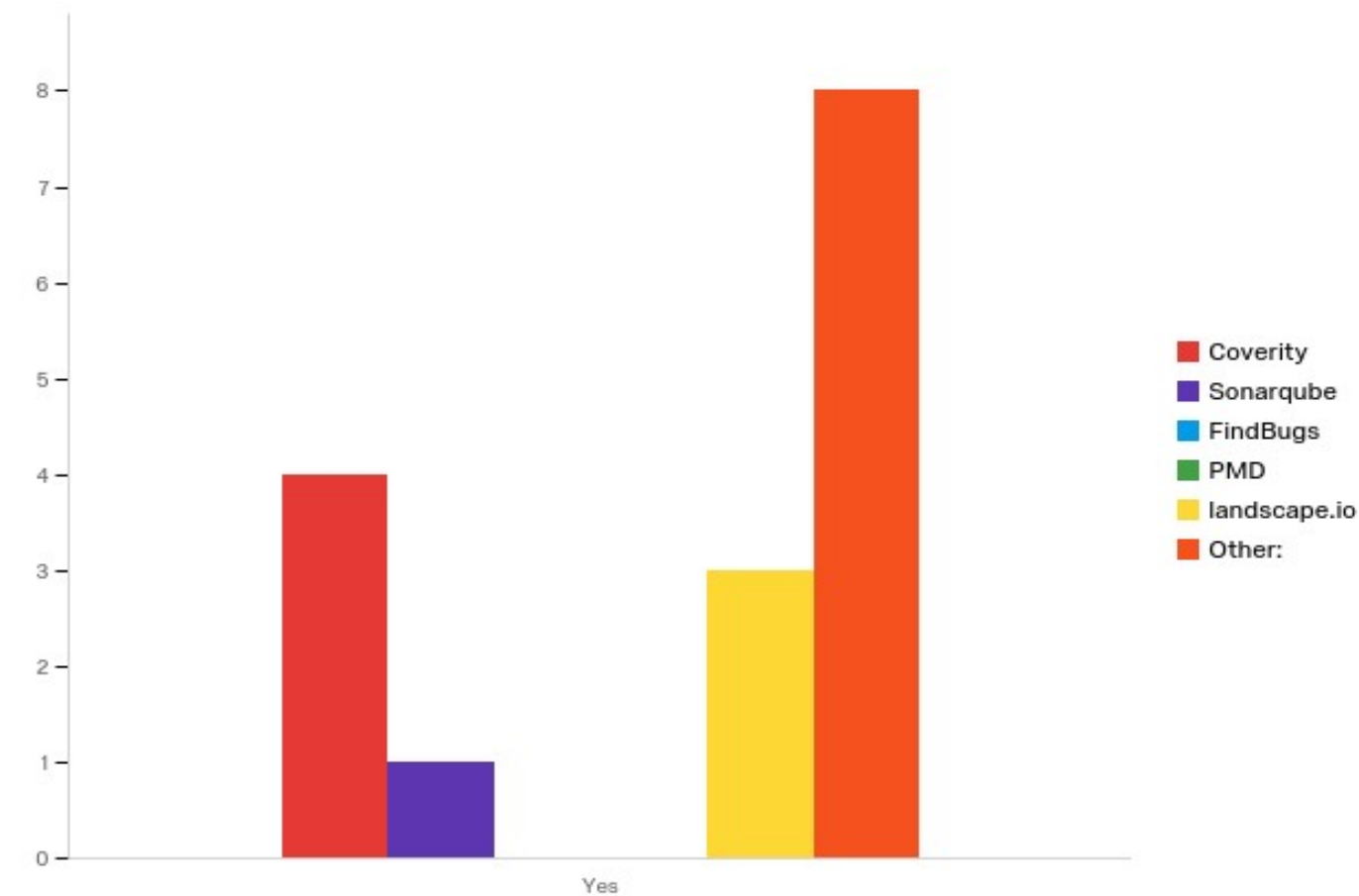
Q4#1_6_TEXT - Other:

Other:
linter-flake8 atom package
Valgrind, Helgrind
valgrind, oprofile
gprof (C++ tool)
flake8
Devel::SmallProf
pylint, pyflakes
pylint
pyflakes, etc
pyinstrument
ANTS



A word cloud featuring various software tools. The words are arranged in a circular pattern, with 'pyflakes', 'valgrind', and 'pylint' being the most prominent in the center. Other visible words include 'package', 'tool', 'c++', 'gprof', 'helgrind', 'oprofile', 'flake8', 'ant', 'smallprof', 'pyinstrument', and 'linter-flake8'.

Q4#2 - Do you currently use it?



#	Question	Yes		Total
1	Coverity	100.00%	4	4
2	Sonarqube	100.00%	1	1
3	FindBugs	0.00%	0	0
4	PMD	0.00%	0	0
5	landscape.io	100.00%	3	3
6	Other:	100.00%	8	8

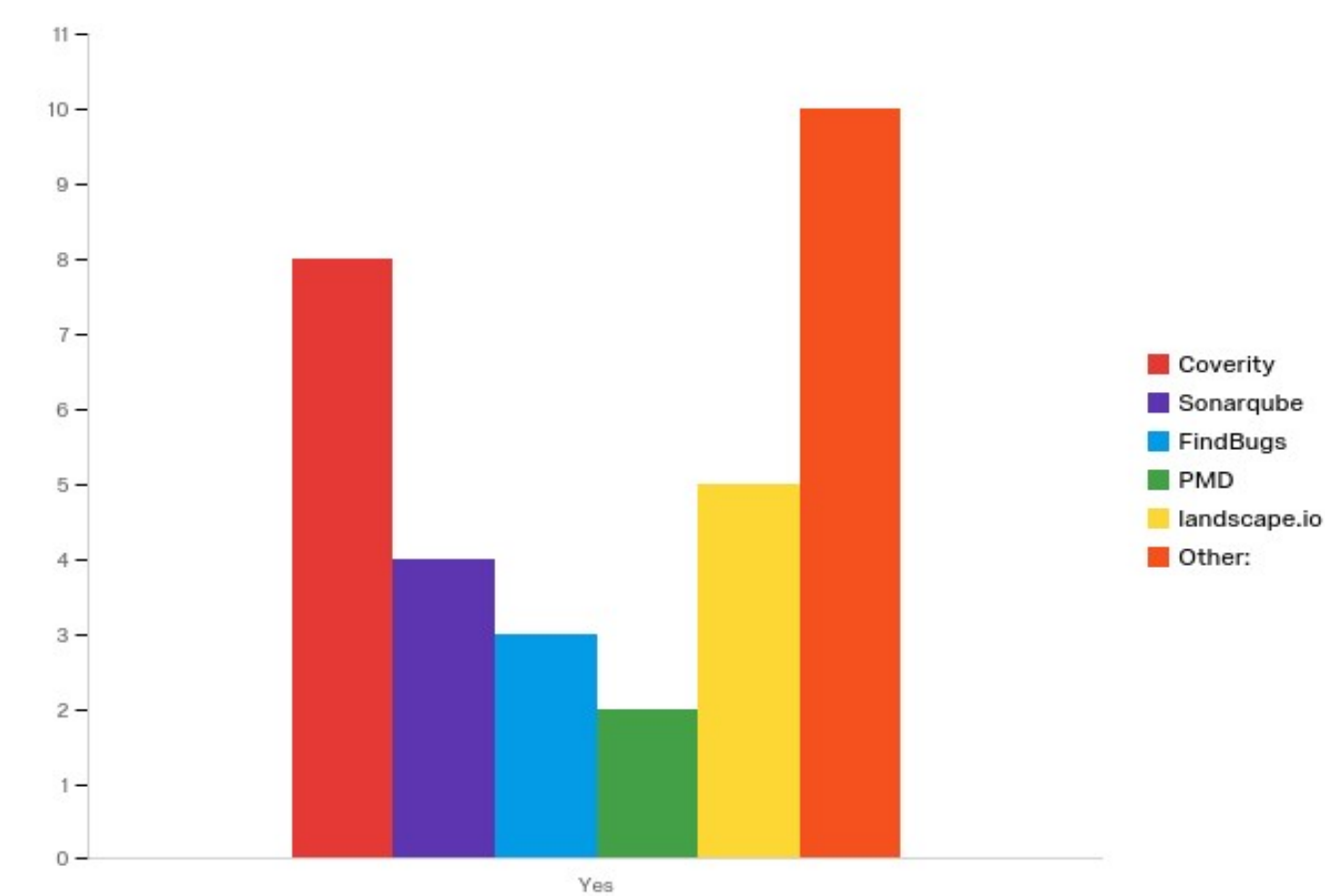
Q4#3_6_TEXT - Other:

Other:

linter-flake8 atom package

Valgrind, Helgrind
valgrind, oprofile
gprof (C++ tool)
flake8
Devel::SmallProf
pylint, pyflakes
pylint
pyflakes, etc
pyinstrument
ANTS

Q4#3 - Would you like to continue/start using it?



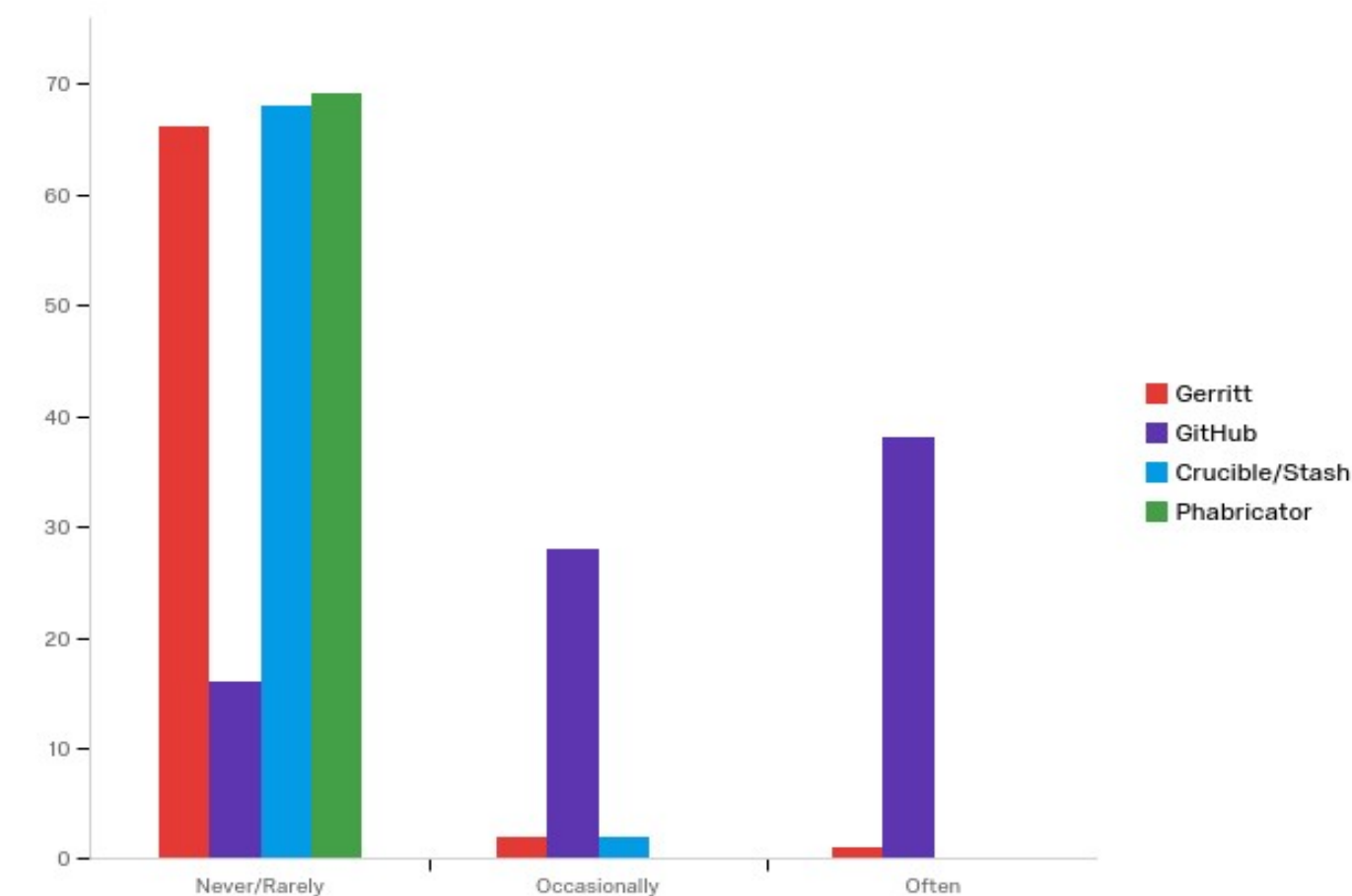
#	Question	Yes		Total
1	Coverity	100.00%	8	8
2	Sonarqube	100.00%	4	4
3	FindBugs	100.00%	3	3
4	PMD	100.00%	2	2
5	landscape.io	100.00%	5	5
6	Other:	100.00%	10	10

Q4#2_6_TEXT - Other:

Other:
linter-flake8 atom package
Valgrind, Helgrind
valgrind, oprofile
gprof (C++ tool)
flake8
Devel::SmallProf
pylint, pyflakes
pylint
pyflakes, etc
pyinstrument
ANTS

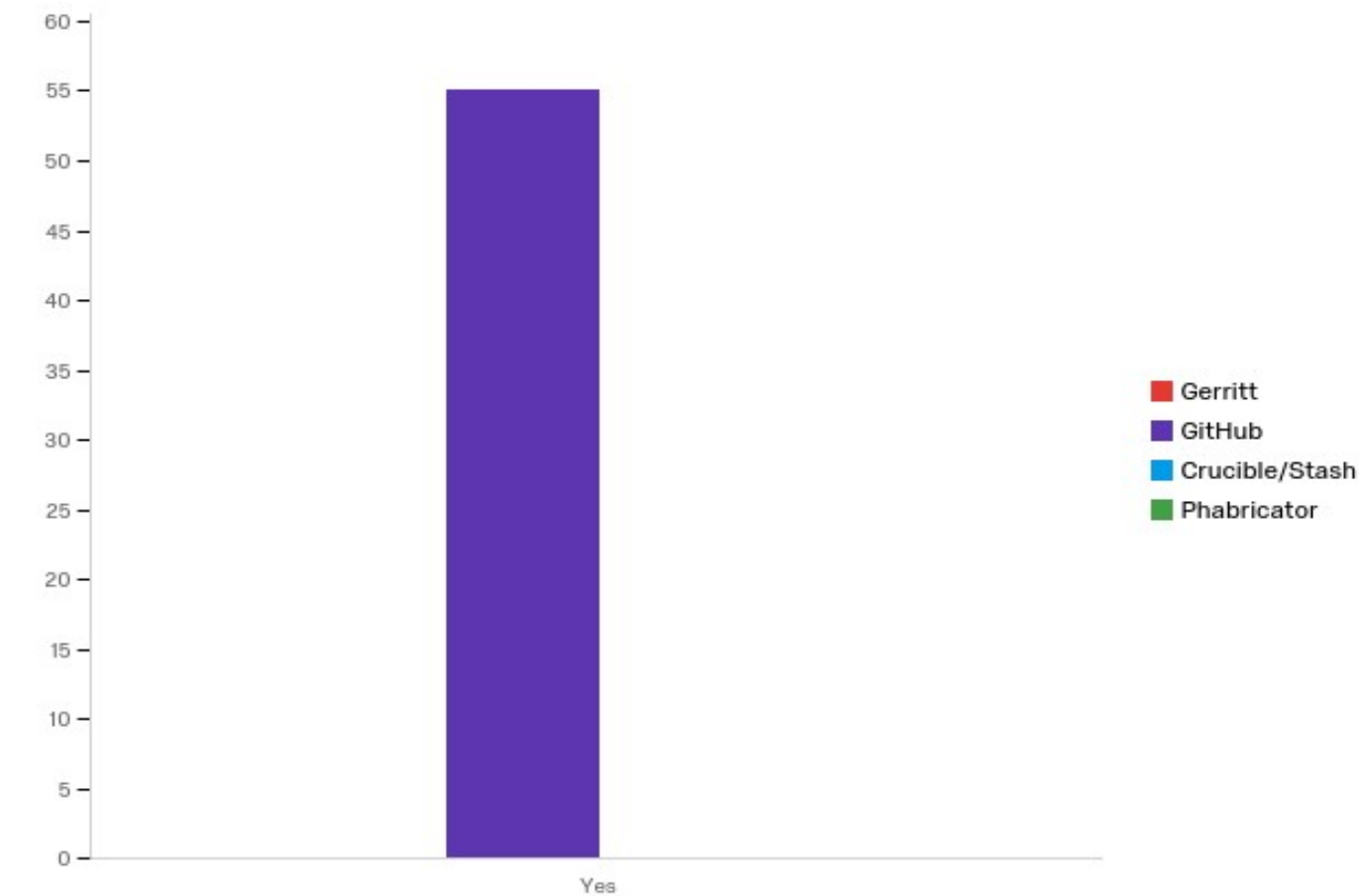
Q5#1 - How often have you used it?

How often have you used the following for code reviews?



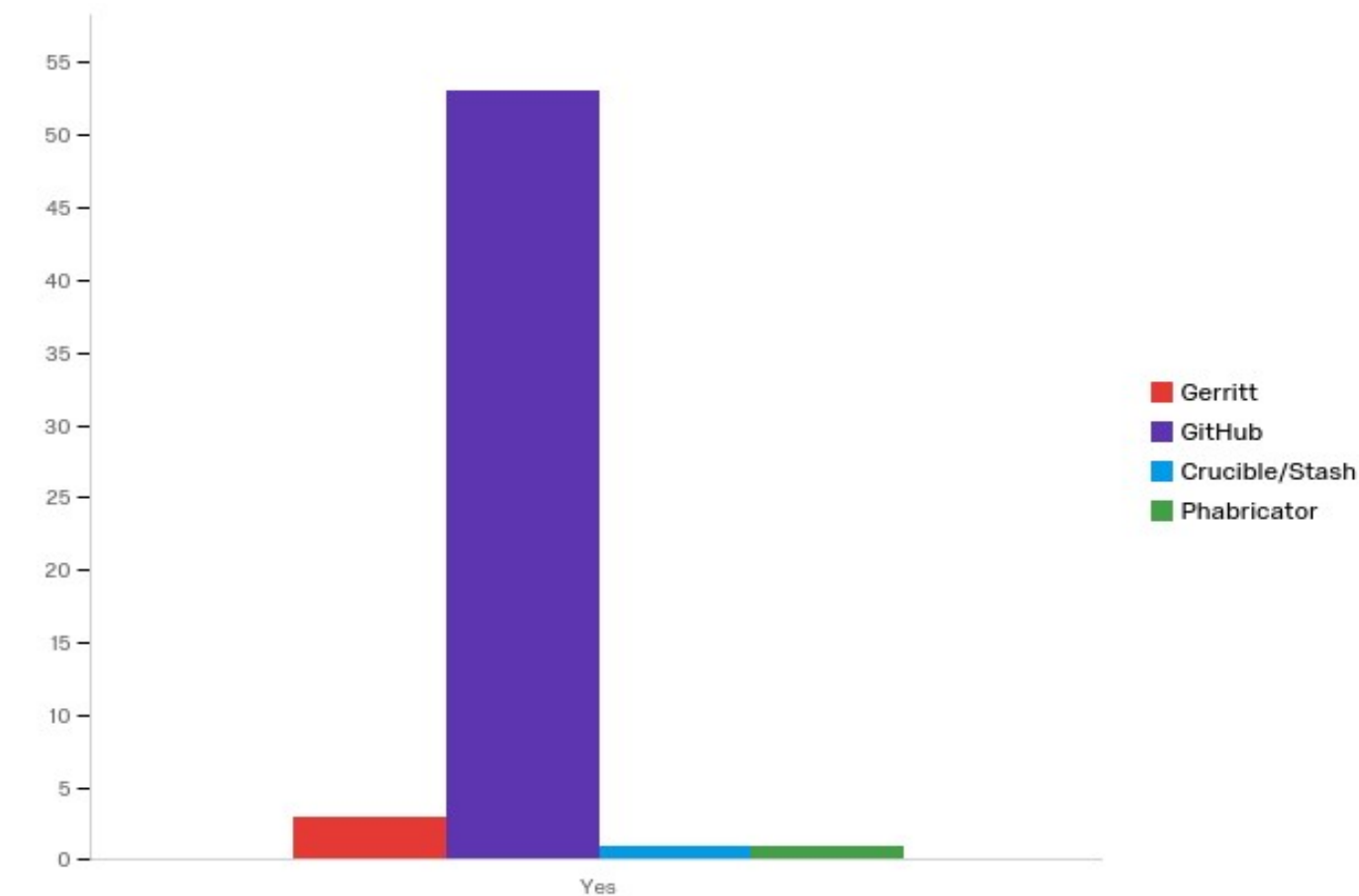
#	Question	Never/Rarely		Occasionally		Often		Total
1	Gerritt	95.65%	66	2.90%	2	1.45%	1	69
2	GitHub	19.51%	16	34.15%	28	46.34%	38	82
3	Crucible/Stash	97.14%	68	2.86%	2	0.00%	0	70
4	Phabricator	100.00%	69	0.00%	0	0.00%	0	69

Q5#2 - Do you currently use it?



#	Question	Yes		Total
1	Gerritt	0.00%	0	0
2	GitHub	100.00%	55	55
3	Crucible/Stash	0.00%	0	0
4	Phabricator	0.00%	0	0

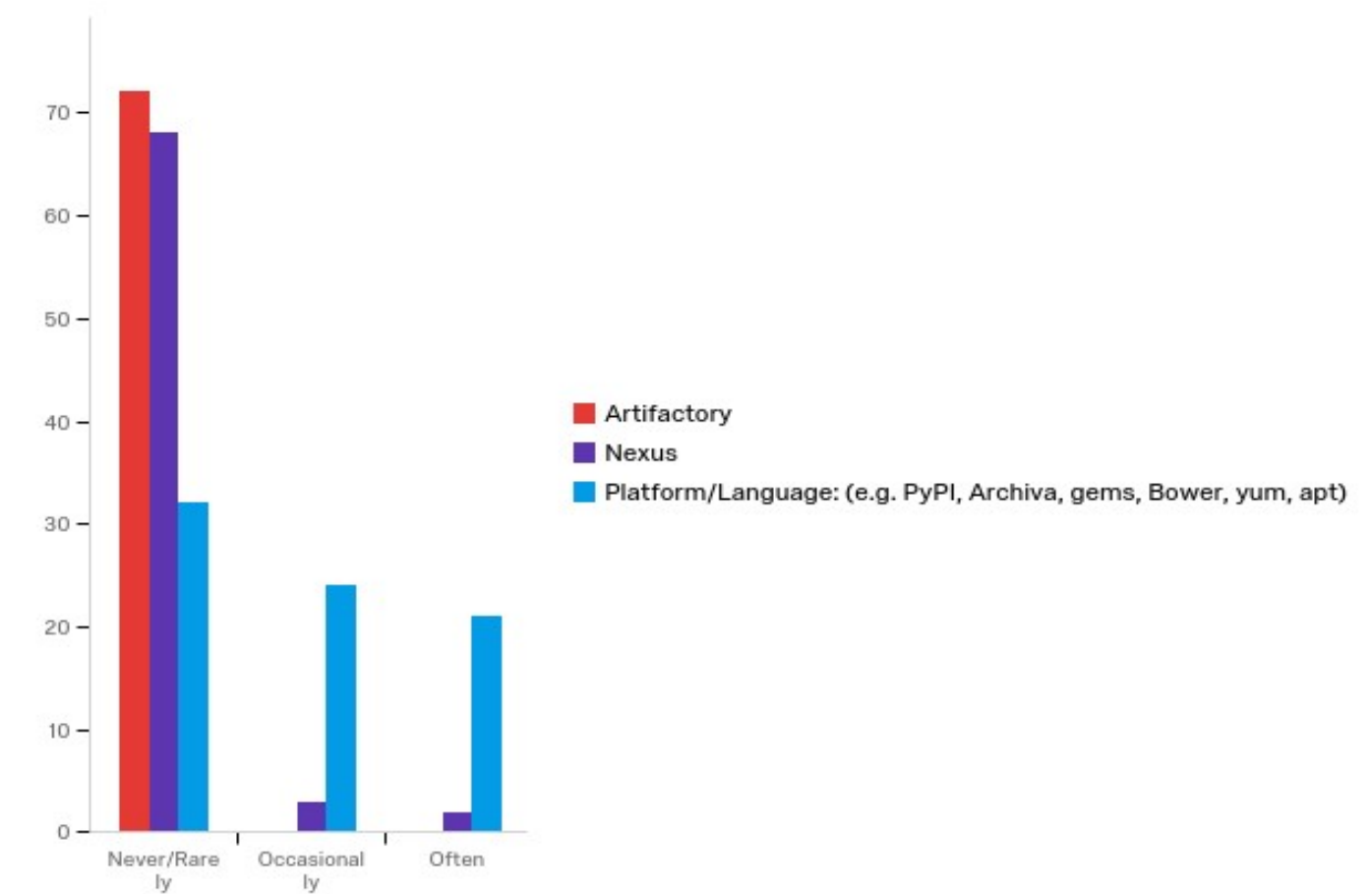
Q5#3 - Would you like to continue/start using it?



#	Question	Yes		Total
1	Gerritt	100.00%	3	3
2	GitHub	100.00%	53	53
3	Crucible/Stash	100.00%	1	1
4	Phabricator	100.00%	1	1

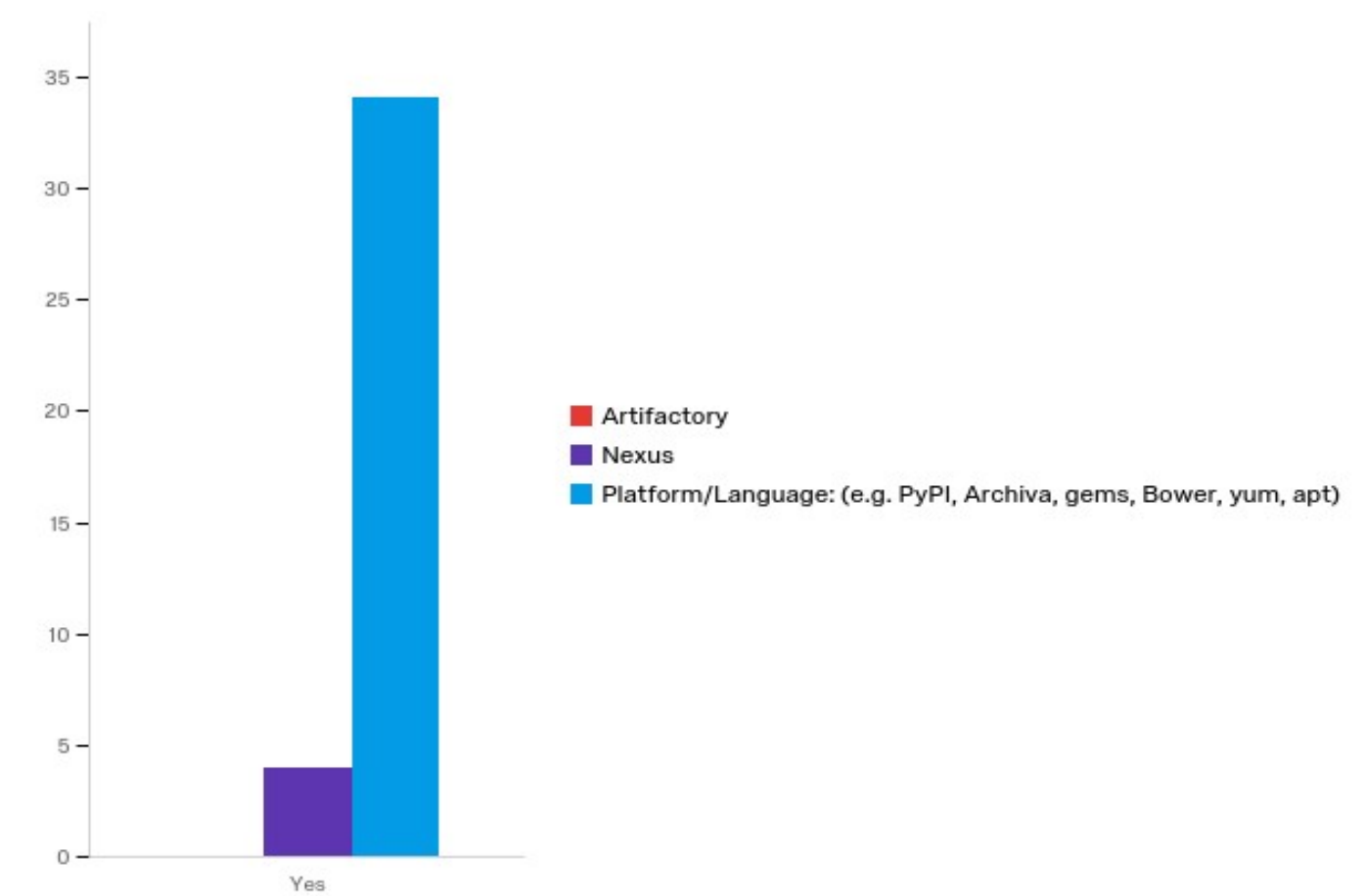
Q6#1 - How often have you used it?

How often have you used Artifact Repositories?



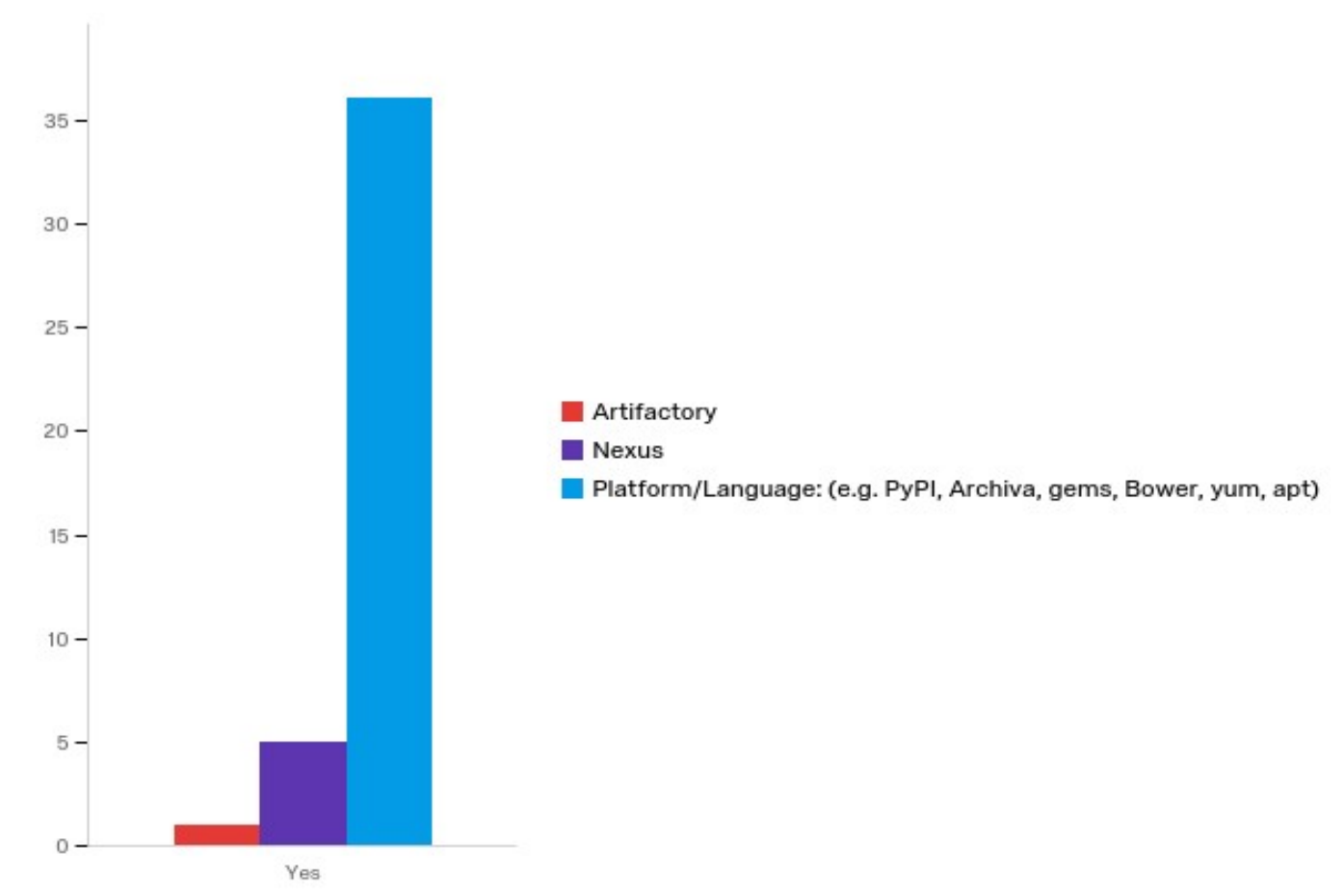
#	Question	Never/Rarely		Occasionally		Often		Total
1	Artifactory	100.00%	72	0.00%	0	0.00%	0	72
2	Nexus	93.15%	68	4.11%	3	2.74%	2	73
3	Platform/Language: (e.g. PyPI, Archiva, gems, Bower, yum, apt)	41.56%	32	31.17%	24	27.27%	21	77

Q6#2 - Do you currently use it?



#	Question	Yes		Total
1	Artifactory	0.00%	0	0
2	Nexus	100.00%	4	4
3	Platform/Language: (e.g. PyPI, Archiva, gems, Bower, yum, apt)	100.00%	34	34

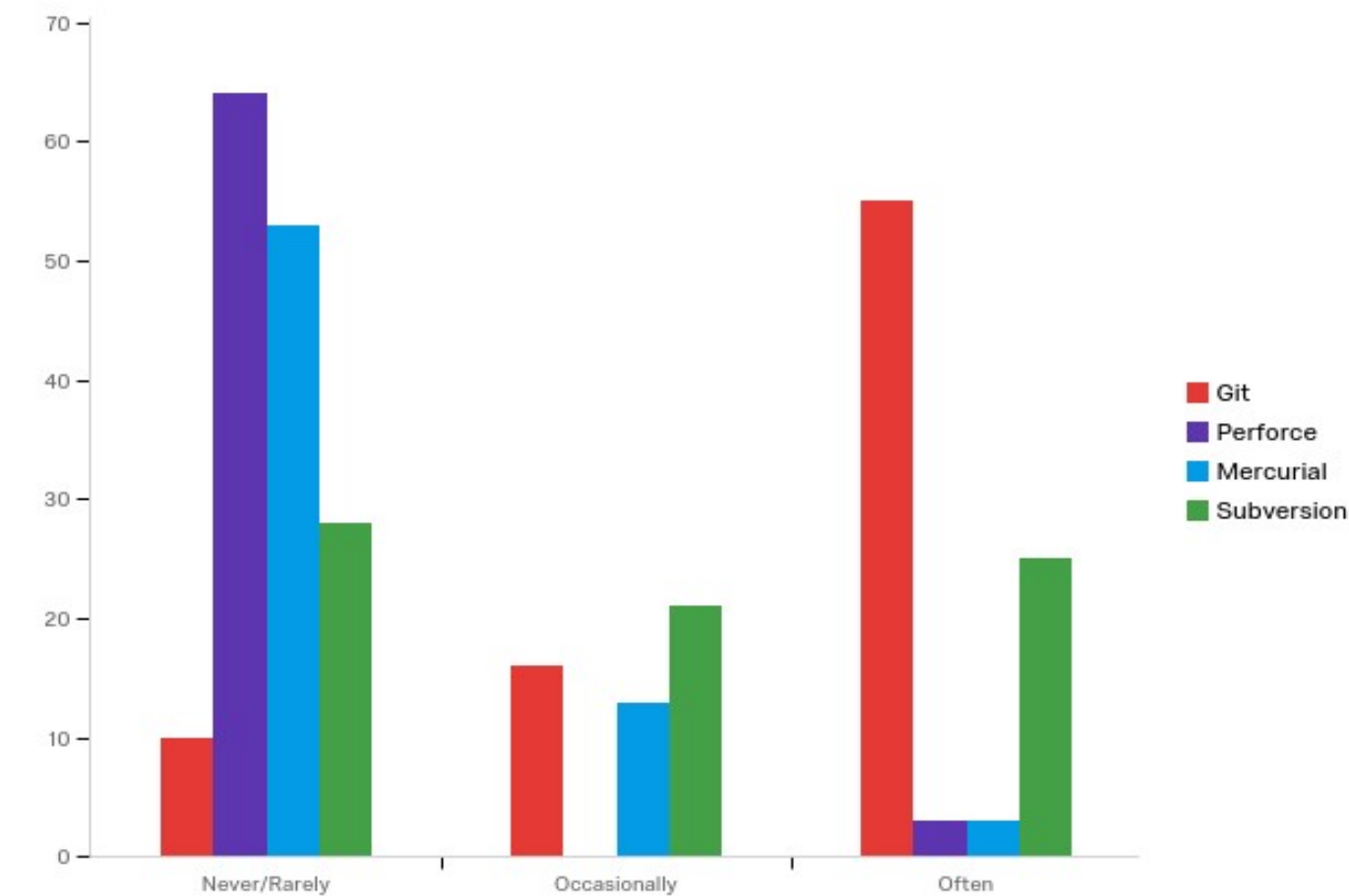
Q6#3 - Would you like to continue/start using it?



#	Question	Yes		Total
1	Artifactory	100.00%	1	1
2	Nexus	100.00%	5	5
3	Platform/Language: (e.g. PyPI, Archiva, gems, Bower, yum, apt)	100.00%	36	36

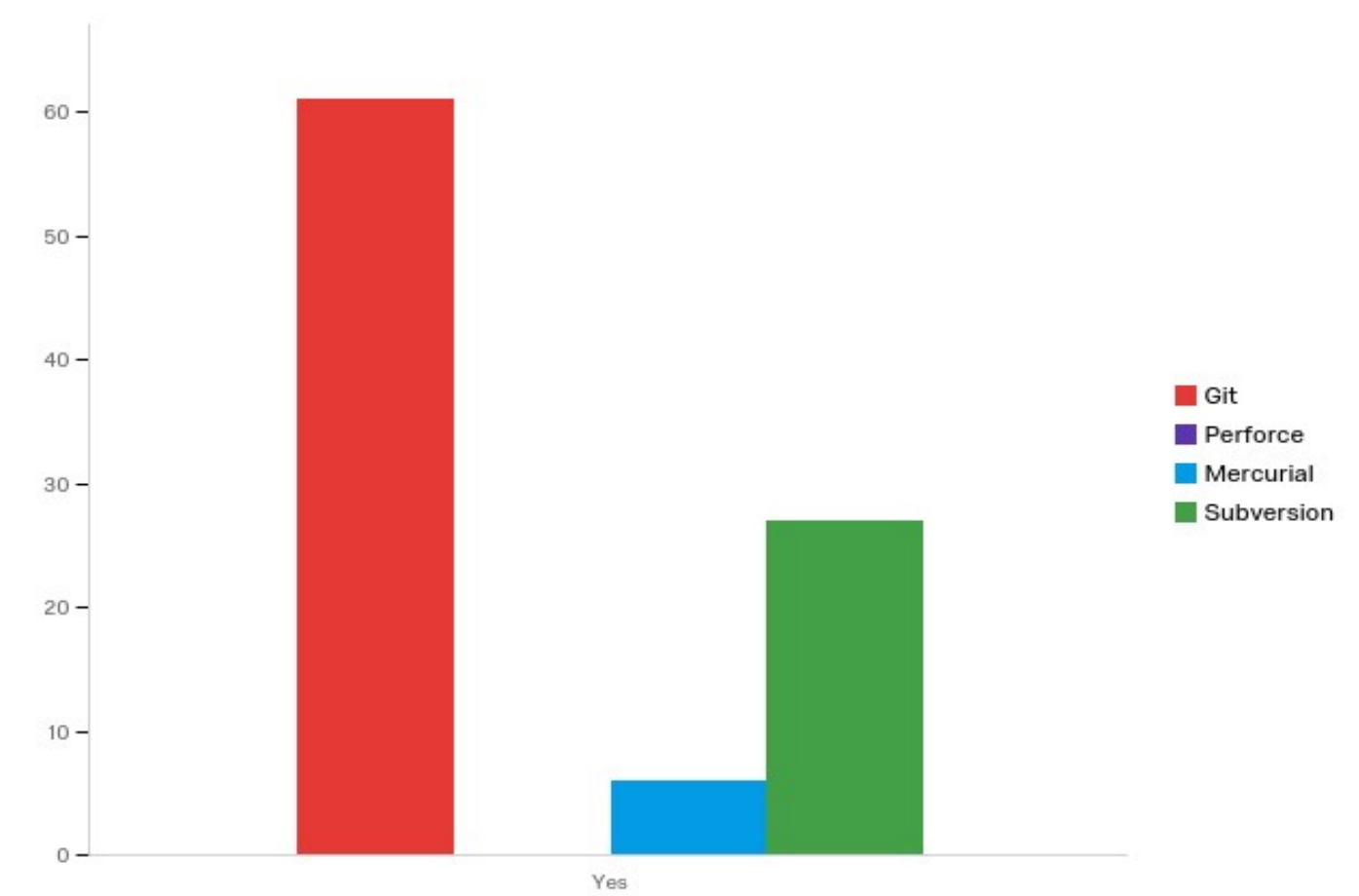
Q7#1 - How often have you used it?

How often have you use these SCMs?



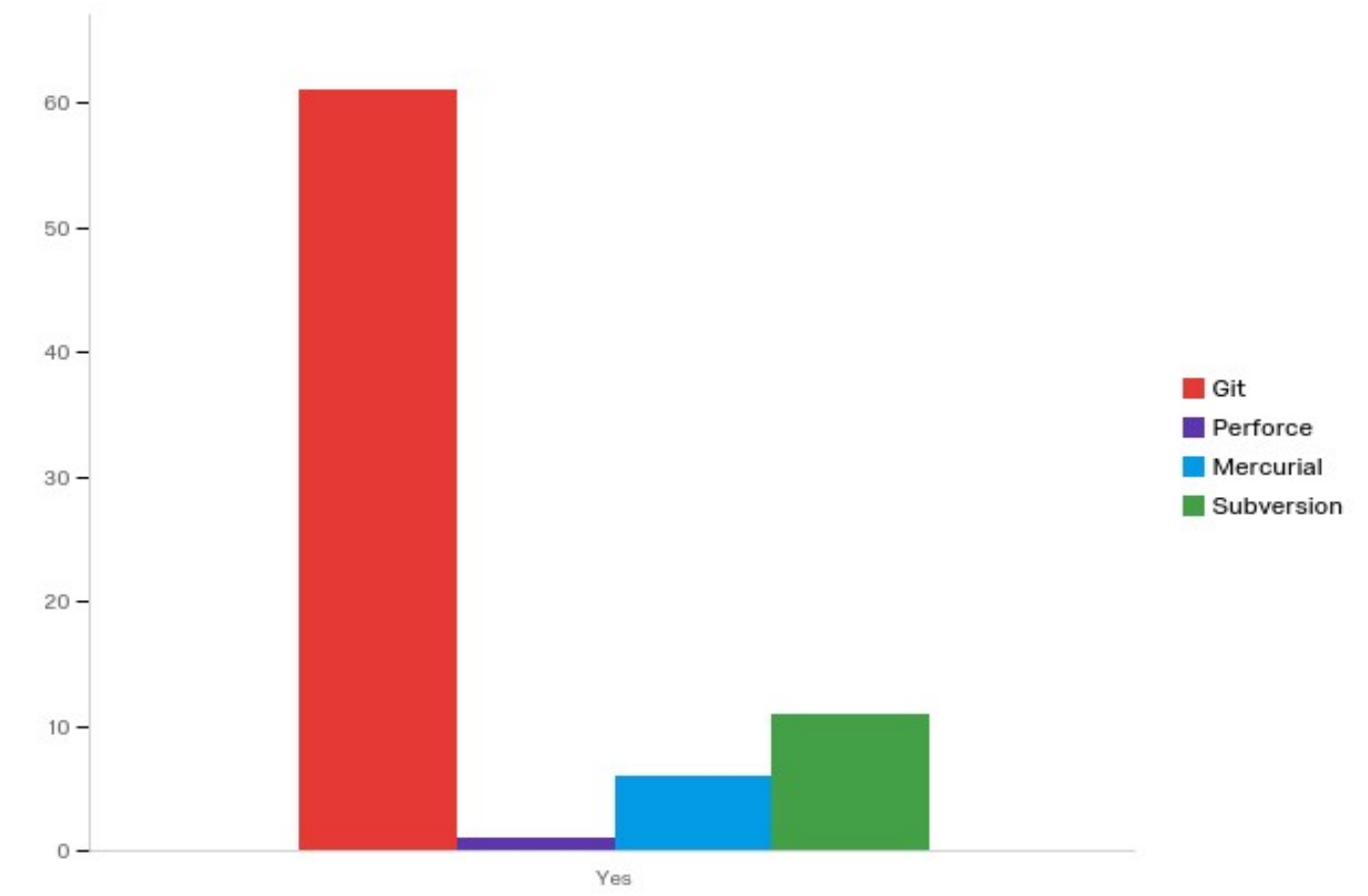
#	Question	Never/Rarely		Occasionally		Often		Total
1	Git	12.35%	10	19.75%	16	67.90%	55	81
2	Perforce	95.52%	64	0.00%	0	4.48%	3	67
3	Mercurial	76.81%	53	18.84%	13	4.35%	3	69
4	Subversion	37.84%	28	28.38%	21	33.78%	25	74

Q7#2 - Do you currently use it?



#	Question	Yes		Total
1	Git	100.00%	61	61
2	Perforce	0.00%	0	0
3	Mercurial	100.00%	6	6
4	Subversion	100.00%	27	27

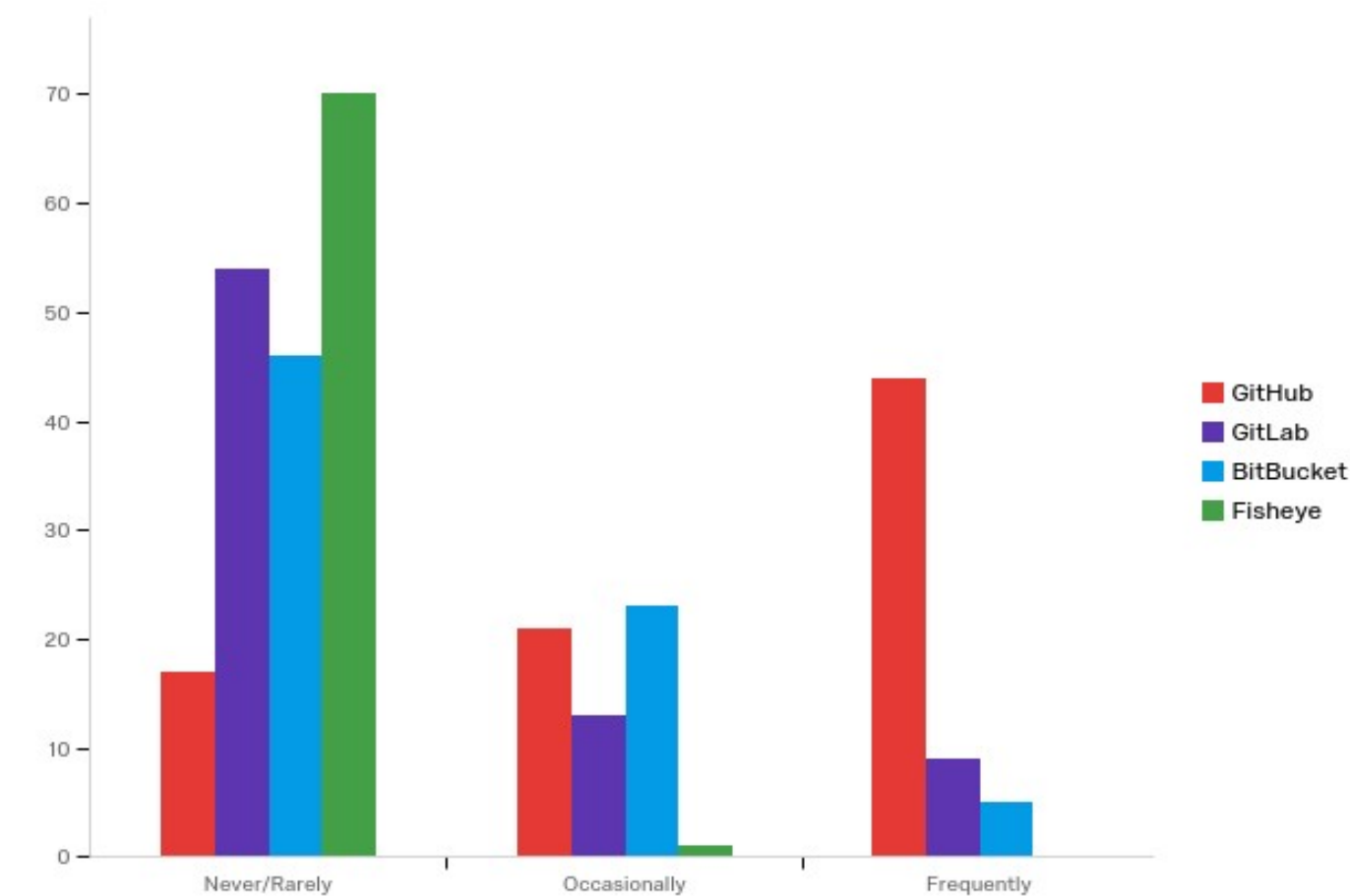
Q7#3 - Would you like to continue/start using it?



#	Question	Yes		Total
1	Git	100.00%	61	61
2	Perforce	100.00%	1	1
3	Mercurial	100.00%	6	6
4	Subversion	100.00%	11	11

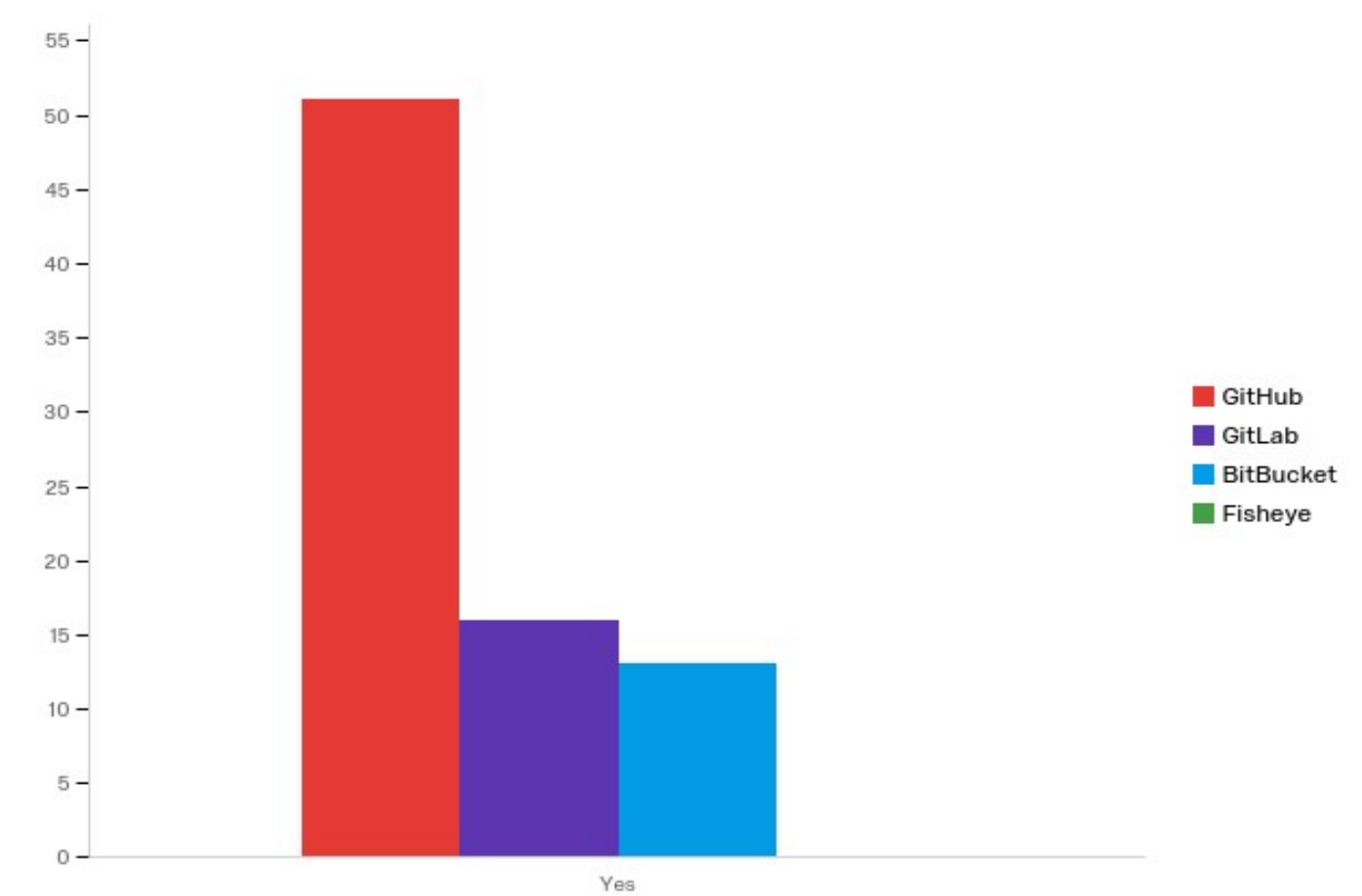
Q8#1 - How often have you used it?

How often have you used these SCM Front Ends?



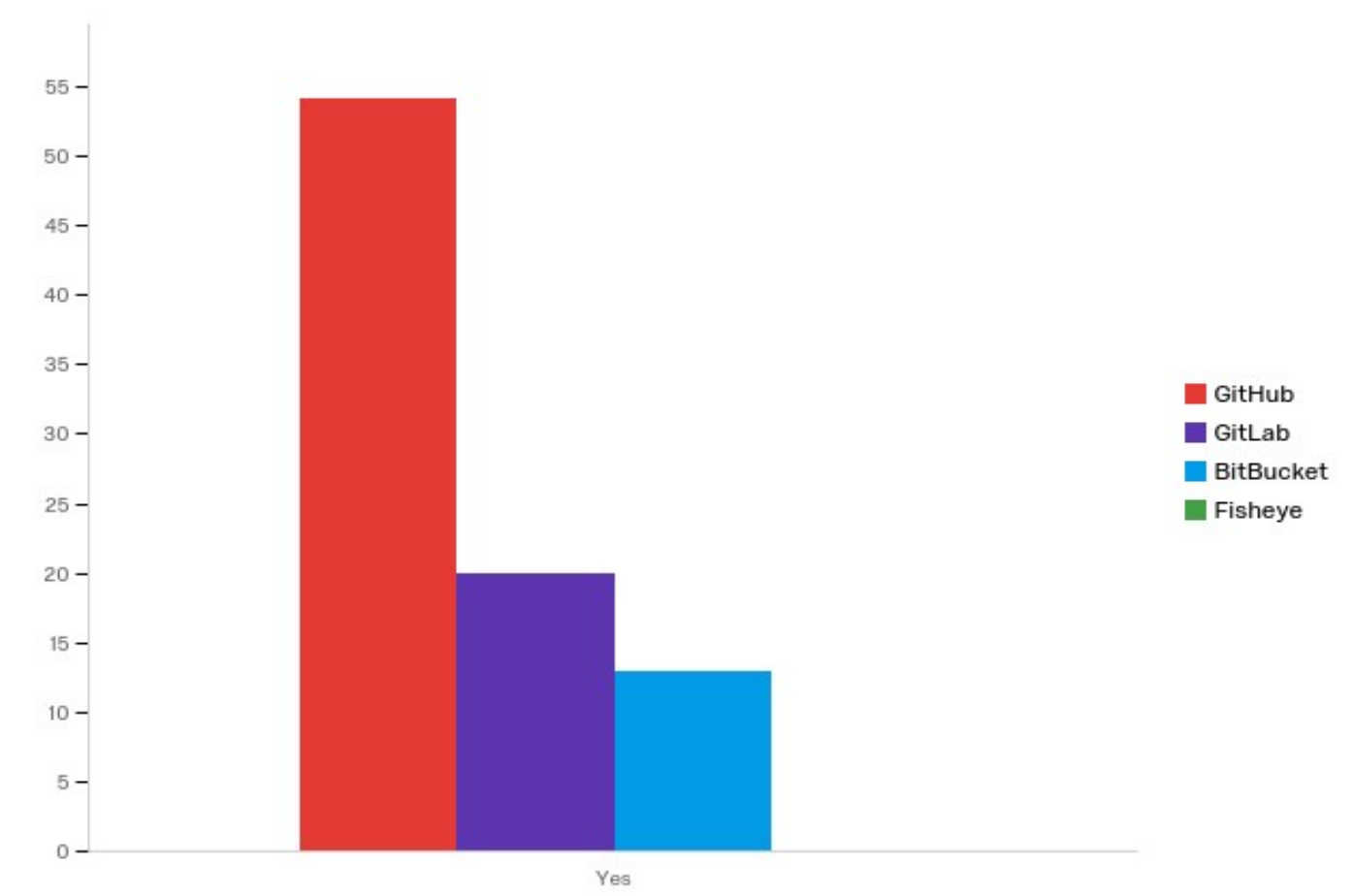
#	Question	Never/Rarely		Occasionally		Frequently		Total
1	GitHub	20.73%	17	25.61%	21	53.66%	44	82
2	GitLab	71.05%	54	17.11%	13	11.84%	9	76
3	BitBucket	62.16%	46	31.08%	23	6.76%	5	74
4	Fisheye	98.59%	70	1.41%	1	0.00%	0	71

Q8#2 - Do you currently use it?



#	Question	Yes		Total
1	GitHub	100.00%	51	51
2	GitLab	100.00%	16	16
3	BitBucket	100.00%	13	13
4	Fisheye	0.00%	0	0

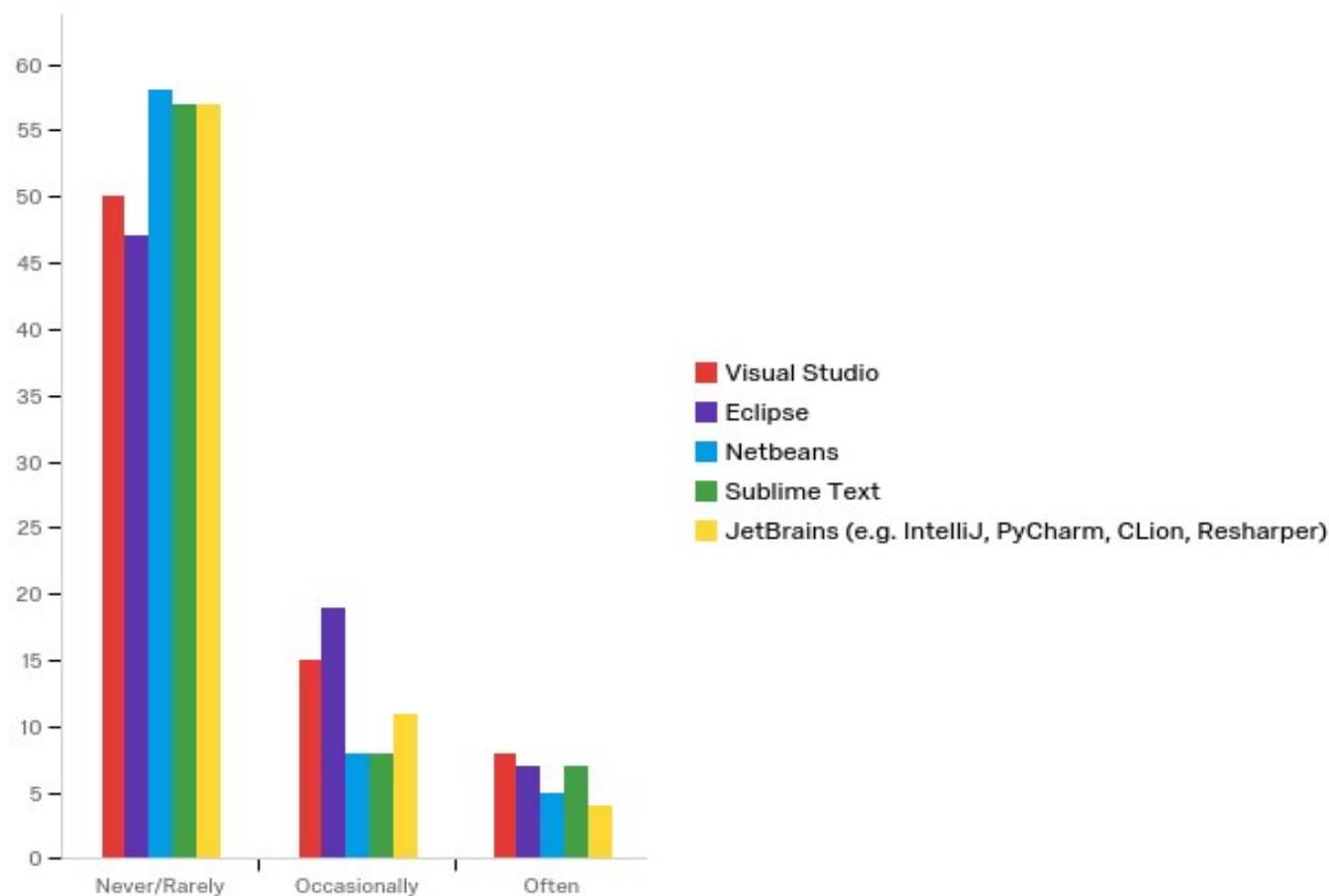
Q8#3 - Would you like to continue/start using it?



#	Question	Yes		Total
1	GitHub	100.00%	54	54
2	GitLab	100.00%	20	20
3	BitBucket	100.00%	13	13
4	Fisheye	0.00%	0	0

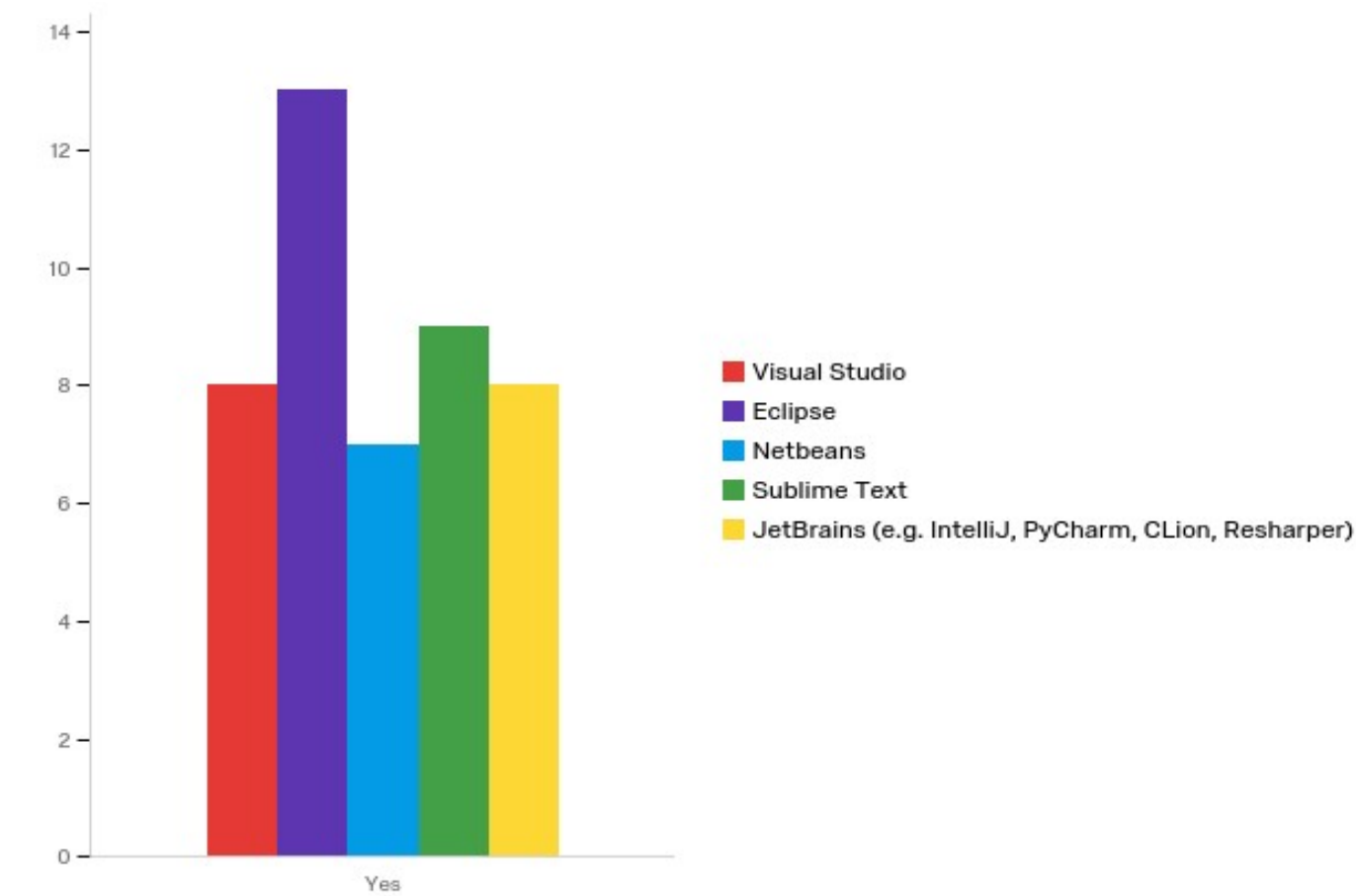
Q9#1 - How often have you used it?

How often have you used these IDEs?



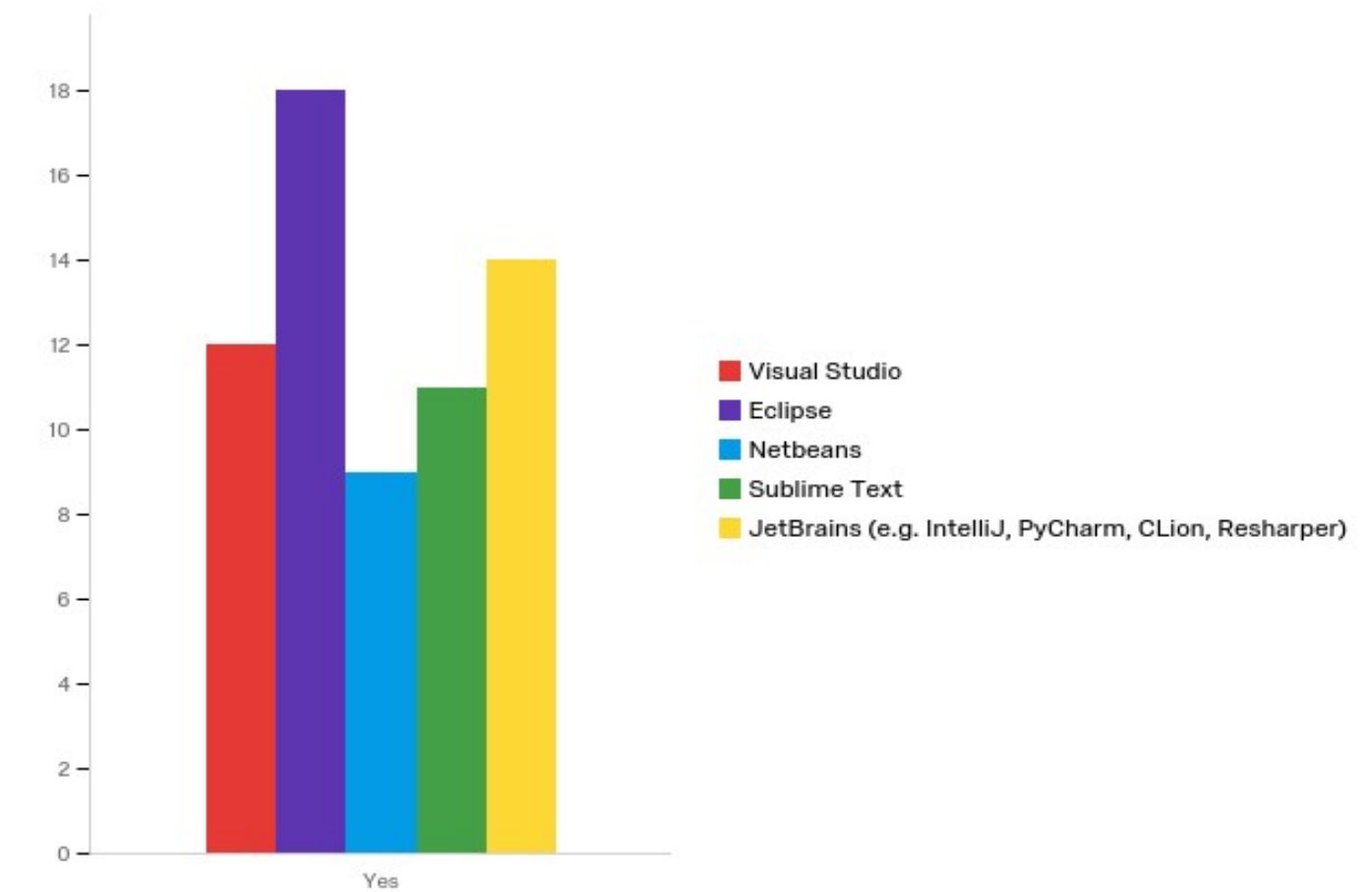
#	Question	Never/Rarely		Occasionally		Often		Total
1	Visual Studio	68.49%	50	20.55%	15	10.96%	8	73
2	Eclipse	64.38%	47	26.03%	19	9.59%	7	73
3	Netbeans	81.69%	58	11.27%	8	7.04%	5	71
4	Sublime Text	79.17%	57	11.11%	8	9.72%	7	72
5	JetBrains (e.g. IntelliJ, PyCharm, CLion, Resharper)	79.17%	57	15.28%	11	5.56%	4	72

Q9#2 - Do you currently use it?



#	Question	Yes		Total
1	Visual Studio	100.00%	8	8
2	Eclipse	100.00%	13	13
3	Netbeans	100.00%	7	7
4	Sublime Text	100.00%	9	9
5	JetBrains (e.g. IntelliJ, PyCharm, CLion, Resharper)	100.00%	8	8

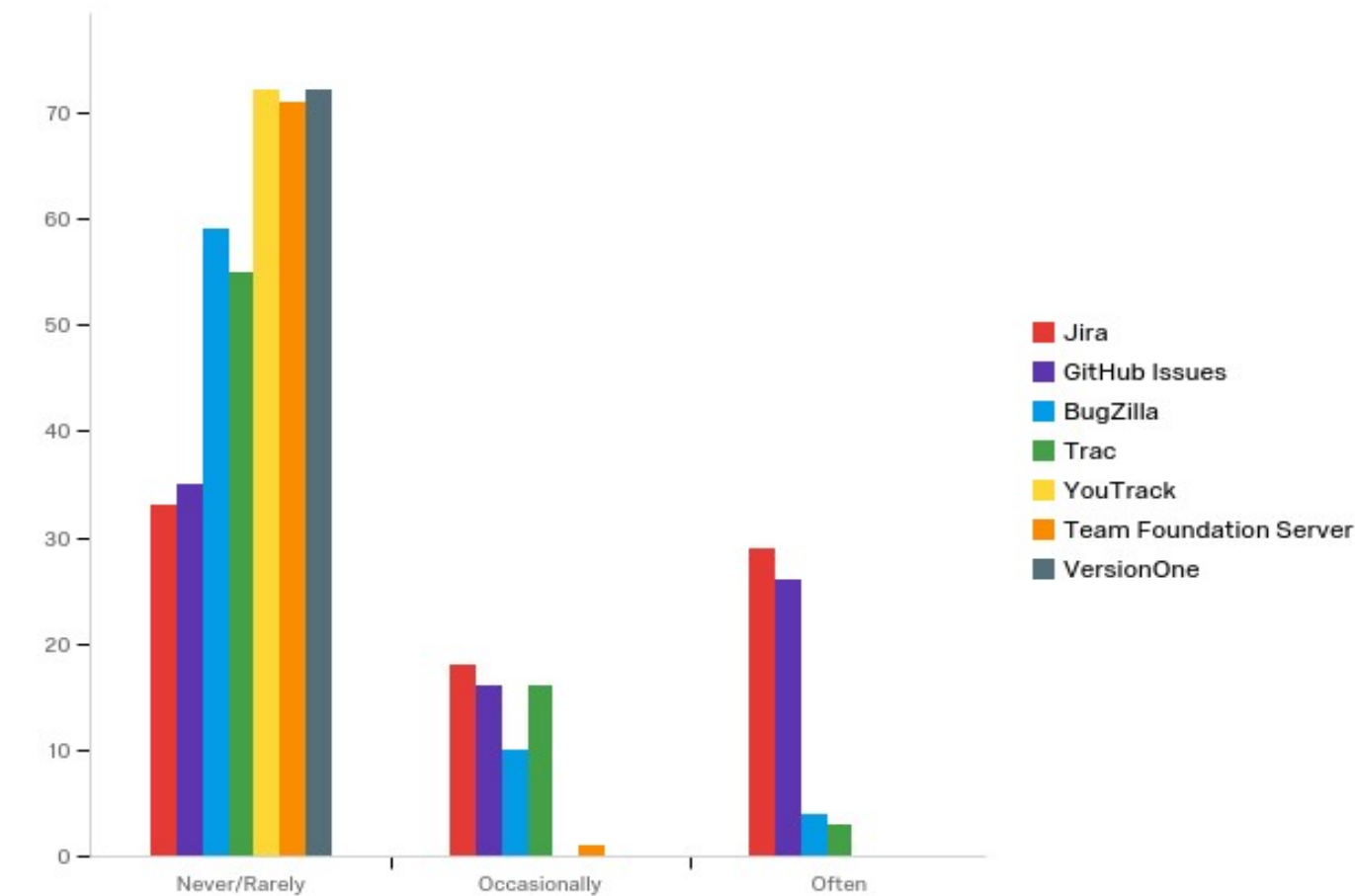
Q9#3 - Would you like to continue/start using it?



#	Question	Yes		Total
1	Visual Studio	100.00%	12	12
2	Eclipse	100.00%	18	18
3	Netbeans	100.00%	9	9
4	Sublime Text	100.00%	11	11
5	JetBrains (e.g. IntelliJ, PyCharm, CLion, Resharper)	100.00%	14	14

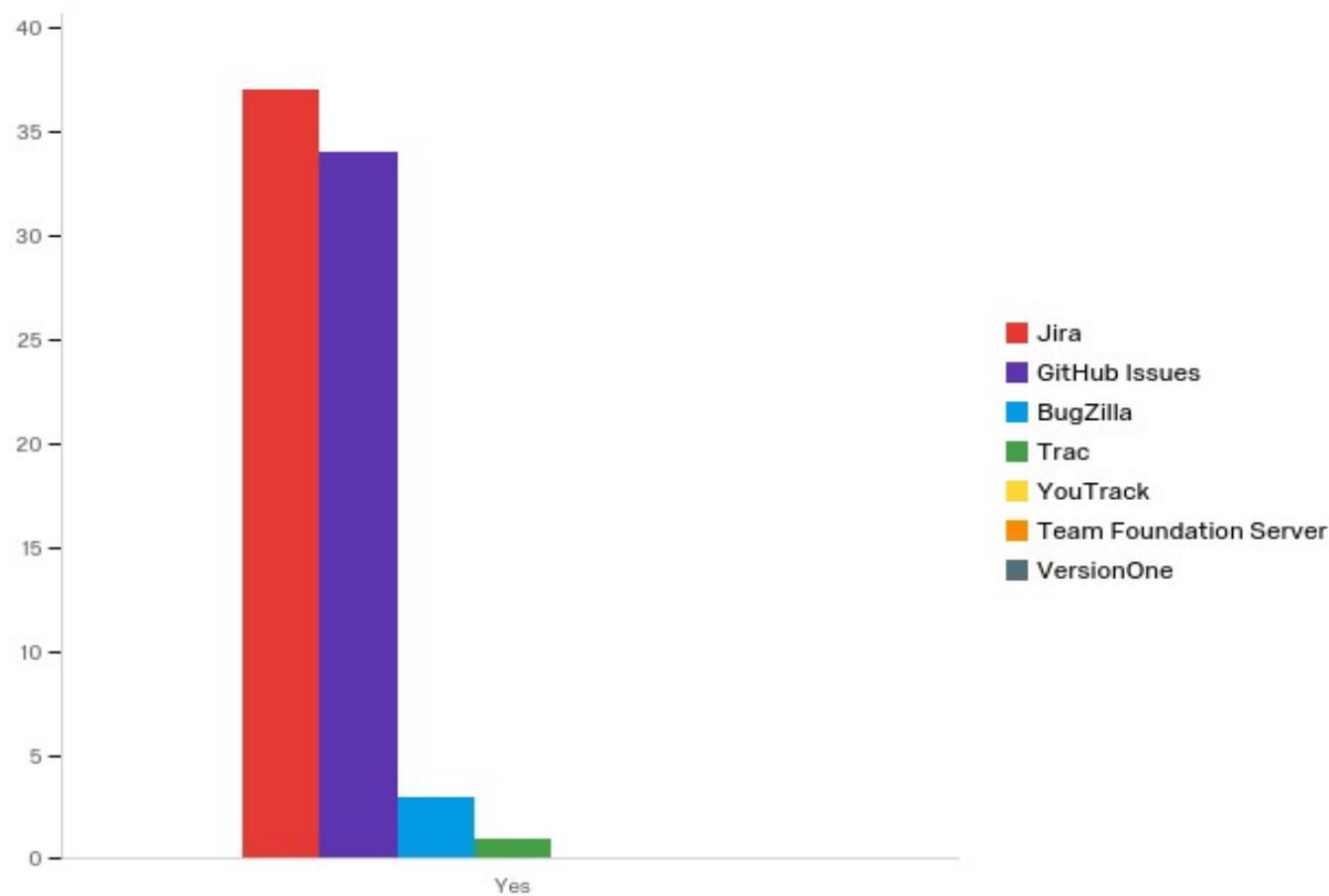
Q10#1 - How often have you used it?

How often have you used these Issue trackers and Agile tools?



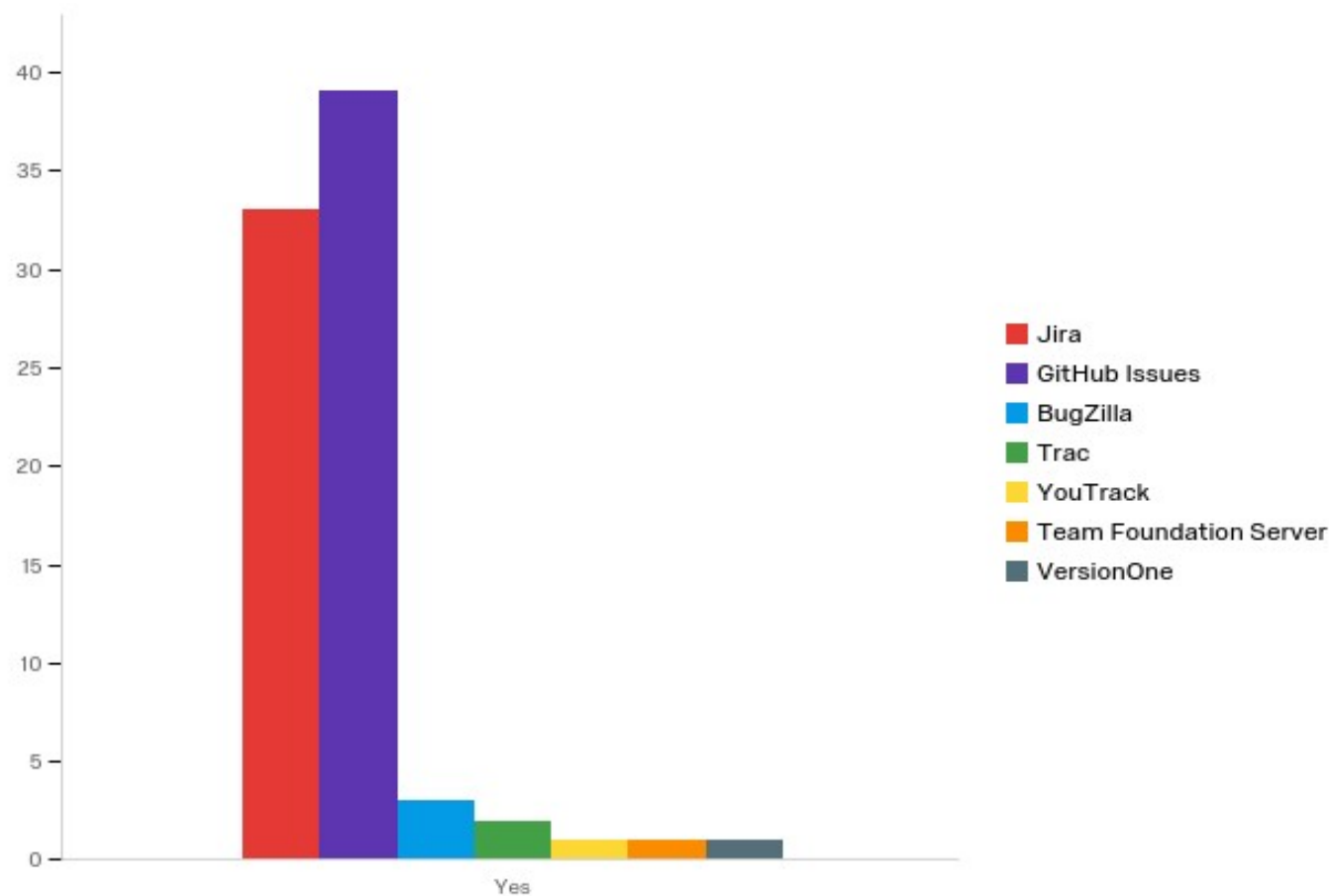
#	Question	Never/Rarely		Occasionally		Often		Total
1	Jira	41.25%	33	22.50%	18	36.25%	29	80
2	GitHub Issues	45.45%	35	20.78%	16	33.77%	26	77
3	BugZilla	80.82%	59	13.70%	10	5.48%	4	73
4	Trac	74.32%	55	21.62%	16	4.05%	3	74
5	YouTrack	100.00%	72	0.00%	0	0.00%	0	72
6	Team Foundation Server	98.61%	71	1.39%	1	0.00%	0	72
7	VersionOne	100.00%	72	0.00%	0	0.00%	0	72

Q10#2 - Do you currently use it?



#	Question	Yes		Total
1	Jira	100.00%	37	37
2	GitHub Issues	100.00%	34	34
3	BugZilla	100.00%	3	3
4	Trac	100.00%	1	1
5	YouTrack	0.00%	0	0
6	Team Foundation Server	0.00%	0	0
7	VersionOne	0.00%	0	0

Q10#3 - Would you like to
continue/start using it?



#	Question	Yes		Total
1	Jira	100.00%	33	33
2	GitHub Issues	100.00%	39	39
3	BugZilla	100.00%	3	3
4	Trac	100.00%	2	2
5	YouTrack	100.00%	1	1
6	Team Foundation Server	100.00%	1	1
7	VersionOne	100.00%	1	1

Q17 - What debugging tools do you use? What debugging tools might you want to use?

What debugging tools do you use? What debugging tools might you want to use...
pydb
gdb, lldb, ipdb, valgrind
gdb through Eclipse. I don't know of anything else applicable to EPICS IOCs, which is not to say that there isn't something. There might be.
gdb, lldb, JVisualVM, Netbeans Profiler
gdb
Would start using whatever tools recommended and locally supported by CDS at LCLS
I use vim python IDE setup, but I'd like a better python vim setup
easy python debugger would be nice
gdb, valgrind, helgrind
gdb, valgrind
valgrind, gdb
gdb, pdb
I use the python debugger and gdb quite often.
It would be very nice to have an easy to use graphical debugger for c++ code.
gdb
lldb, gdb
gdb
For java/jsp I use NetBeans. For C/C++ I use gdb, especially gdb within emacs. I'm happy with those tools for those languages.
LLDB, IgProf
gdb, valgrind, pdb
built-in language elements, gdb, profiling, ganglia(!). Open to suggestion.
Careful reading, occasional diagnostic print statements. Better debugging tools lead to poor coding.
gdb & idb
Visual Studio, Powershell ISE, Dreamweaver, InfoPath Designer
XCode, Valgrind.
Matlab debugger. Simple print statements in Python.
gdb, pdb

eclipse cross platform debugging, wireshark
Perl debugger
PyCharm, Matlab, EMACs, Eclipse built-in functions as well as some custom scripts
pdb, printf logging
gdb, valgrind
pdb, visual studio, xcode. I'd like a visual debugger for python, but haven't figured out what that would be.
gdb
gdb, pdb in python
pdb, gdb
In the end print statements. I should learn how to use Netbean's debugger at some point.
gdb, pydb, valgrind
Typically I just use a linter. I would be interested in using more sophisticated tools, however these typically require some kind of subscription/purchase which isn't necessarily straightforward to justify.
gdb, a little pdb, profilers
Currently using pdb for python. For C/C++ I occasionally use gdb and/or ddd. For networking, wireshark and tcpdump.
pdb gdb valgrind
gdb
printf, and rarely gdb
various debugging packages for vim (syntastic and associated things; old school like that)
netbeans, visualvm, print statements
None really, I'd like to learn about what's out there and how to use them!
Currently very few permanent debugging tools for PCDS, but making a strong push to have pytest for most of the major modules during the 6-month down. Would be very interested to see Jira / Github integration
Visual Studio IDE tools (Step into, over, TwinCAT single pass, etc)
Gdb and Pdb for Python stuff
gdb valgrind

A word cloud featuring various debugging tools and programming languages. The words are arranged in a roughly circular shape, with some words being significantly larger than others. The colors are in shades of teal and green.

debugger
valgrind
gdb
pdb
python
tool
statement
visual
better
vim
nice
printf
interested
netbeans
language
built-in
matlab
lldb
easy
print
ide
c/c++
emacs
eclipse
studio
netbean

Q14 - Have we missed anything? Any additional comments? (Optional)

Have we missed anything? Any additional comments? (Optional)
I'm not a hard-core developer currently, rather I work on documentation for LSST, and I write mostly in rst right now, so take this as you will as far as usefulness..!
For operations: Inspec
Unit test coverage reports? E.g., codecov or coveralls.
Most of the things I never/rarely use are things I've never heard of. Maybe some of them do things I would find useful if I knew about them?
KIPAC/Campus user
For programming PLC's, we have a wide variety of software to support vendor specific hardware. Rockwell Design Studio 5000, Codesys, etc.
I selected "GitHub" for "code review" and Project Management, but I really meant "GitLab" (there was no option in the survey.
please discourage the use of CVS
<p>I listed Make as a tool I occasionally use. There are two reasons for this:</p> <ul style="list-style-type: none">* building someone else's software which uses make* maintaining legacy Qt apps which use qmake/make <p>I would never use Make for a new package I was writing.</p>
<p>In cases where I already use a tool in some category regularly and find it adequate, I generally don't have much interest in learning about other tools for the same task. Some of them might be better than whatever I'm using, but it's hard to see how they could be enough better to justify the overhead of learning (and remembering) how to use yet another tool.</p>
<p>I write small-scale programs that are essential to my scientific work. As such, I am too small a fry to get any support or training at SLAC. So I am still stuck in the 20th century. There are many people like me at the lab. Will you remedy this?</p>
OS familiarity and management
<p>Missing Questions:</p> <p>Where do you get your code tools/development support?</p> <p>What support should be provided in-house?</p> <p>Should there be SLAC-wide code and tool standardization policies, what should they be?</p>
Under IDEs, you didn't have a column for XCode. This is my principle IDE.
CVS for code repository.
You are missing CVS under Source Code Management.
You are missing CATER under "Project Managements (Agile), Development Workflows, CCBs, and Defects".
It would be very nice to get some training and exposed more to new tools.
you could add a "would like to know more" column to some of the above selections if you care about that. e.g. I haven't used any of the static analysis tools, but I would be interested in doing more static analysis than I have

done yet.
Yes, you've totally missed a class of the run-time profiling tools, such as: oProfile, Valgrind, etc.
You may also add the OO/UML modeling tools
trellio for project management
I consider emacs with various plugins for python development superior to any IDEs.
Most of the names I've never heard of got marked as 'Never' but only because it's not clear if starting to use one of them in place of what I already use would be an improvement.
Thanks for the survey.
Code reviews on github suck. CodeCollaborator is a much nicer tool, but \$\$\$... Even Gerrit is better.
GITHUB
I am on a mission to get LCLS to convert to github. Having been a long time user, I know we are missing out on the support for collaborative development github provides. Would be great to have it as a cross-lab standard.
I use GitLab CI instead of any of the ones listed above.
editors emacs, vim
profiling
We currently use buildbot.
Some of those tools I'm not familiar with, the Artifact and code review, I don't know if I'd want to start using them.
There is probably a lot of distributed expertise in the different divisions at the lab, and in different experiments/collaborations. It would be great to learn from and build on that. Any attempt to provide some centralized alternative (like servicenow instead of JIRA!) would likely be a disaster.
I'm a PLC engineer (or at least a big part of my job is) which is a subject this survey doesn't touch on much.