2018 NODE.JS USER STUDY
DETAILED REPORT OF FINDINGS

May 2018

THE LINUX FOUNDATION

Node.js is emerging as a universal development framework for digital transformation with a broad diversity of applications.

With nearly 10 million Node.js users, three in four users are planning to increase their use of Node.js in the next 12 months. Node users continue to report positive business impact, including improving developer productivity and satisfaction and lowering cost.

Major highlights

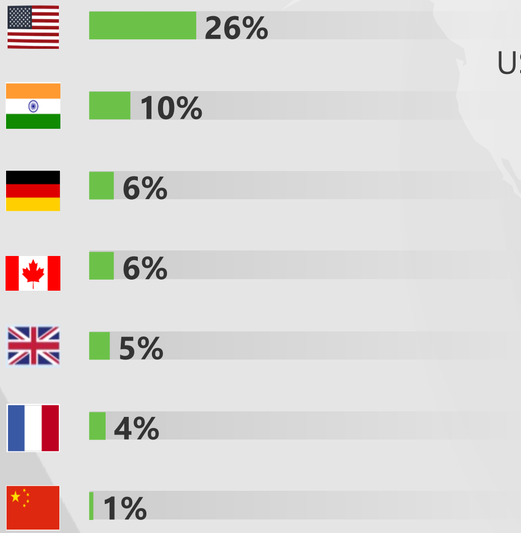
1. Node.js is continuing to have a positive impact on users particularly around developer productivity and satisfaction; when asked to describe Node.js, respondents use mostly positive terms like – “fast”, “easy”, “awesome”, “powerful”, “flexible” and even “fun”.
2. The coming year will likely see continued growth for Node.js. Most users expect to increase their usage, particularly in Latin America and EMEA.
3. A differentiated strategy by region may be called for given Node’s global presence and the varying needs and approaches by region. Among other things, there are key differences in business profile, deployment locations, language & resource usage, priorities (e.g., around package managers or LTS), educational needs and experiences, all of which may call for different approaches worldwide.
4. It is becoming increasingly important to users to manage different packages for multiple environments – but having access to multiple registries is not particularly important outside Latin America. npm is by far the most widely used package manager, but Yarn is gaining in popularity in many segments.
5. Recent improvements around education have registered with users as evidenced by improved scores for ease of learning Node.js, and for the availability and quality of resources. But more could be done, particularly in meeting the needs of some segments (APAC, EMEA and new users) and in some topic areas (Managing Node.js in Production and Node.js and Security).

International presence

Collectively, respondents span **100+** countries and speak at least **60** languages

Fewer than half consider English their primary language

WHERE RESIDE *top mentions*



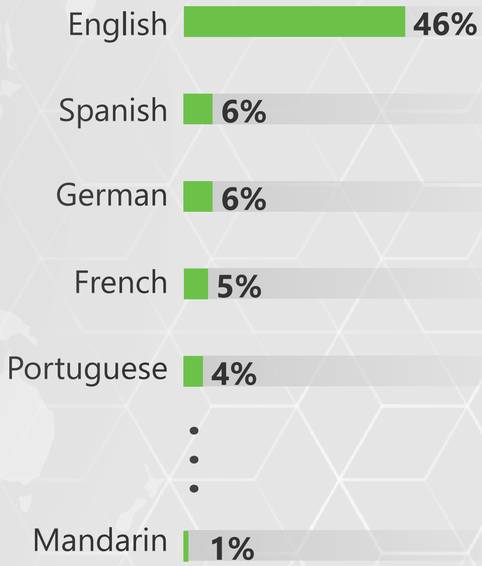
US / Canada
32%

EMEA
44%

APAC
18%

Latin / South America
7%

PRIMARY LANGUAGE SPOKEN *top mentions*



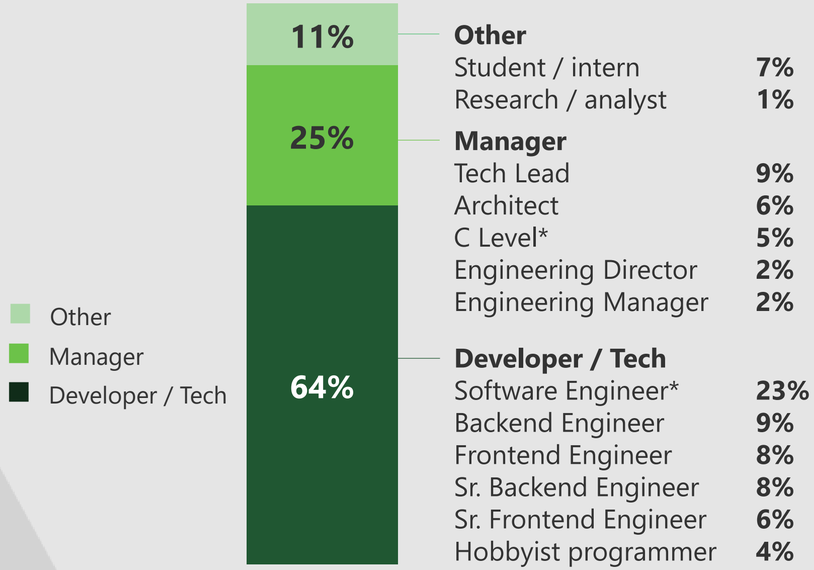
SOURCE: Q4, Q7

Professional profile

Typical respondent is a developer in a small (<100 employees) firm, with 5 years professional experience

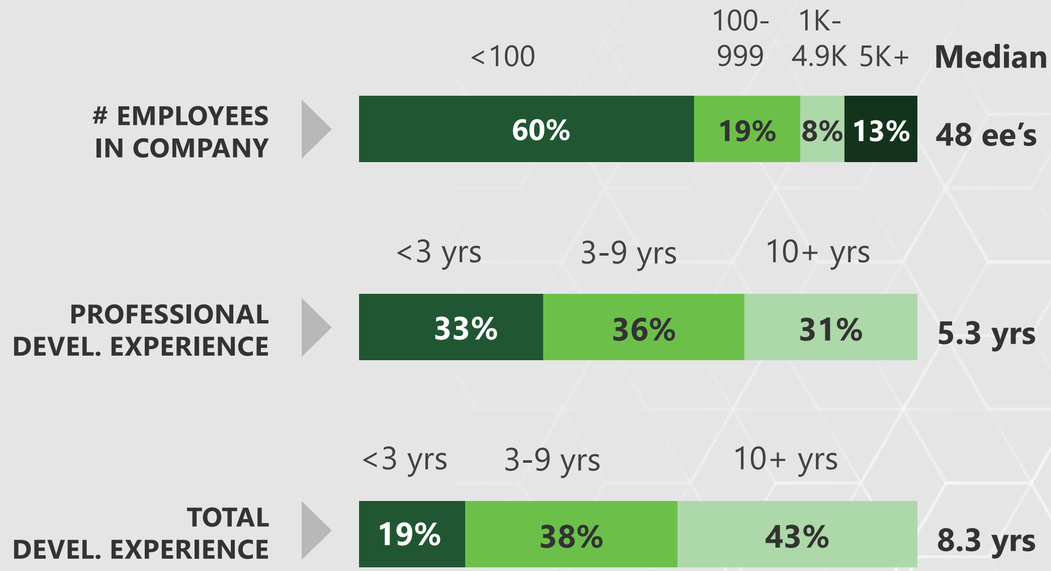
Although many have 10+ years total development experience, respondents are somewhat less experienced overall in this wave

PROFESSIONAL TITLE



Other	
Student / intern	**7%**
Research / analyst	**1%**
Manager	
Tech Lead	**9%**
Architect	**6%**
C Level*	**5%**
Engineering Director	**2%**
Engineering Manager	**2%**
Developer / Tech	
Software Engineer*	**23%**
Backend Engineer	**9%**
Frontend Engineer	**8%**
Sr. Backend Engineer	**8%**
Sr. Frontend Engineer	**6%**
Hobbyist programmer	**4%**

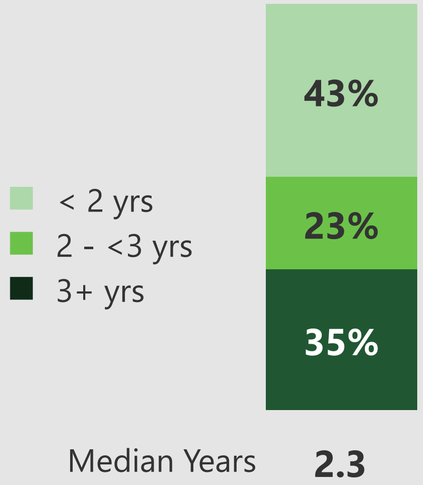
DEMOGRAPHICS



The typical respondent has been using Node.js for just over 2 years, and spends more than half of their development time with Node.js

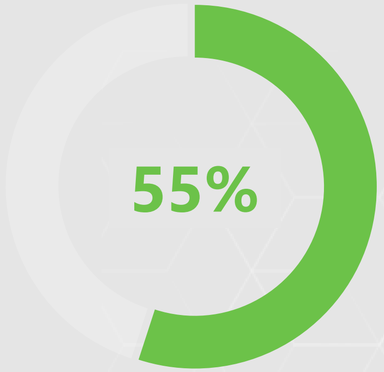
Experience with Node.js

YEARS USING NODE.JS

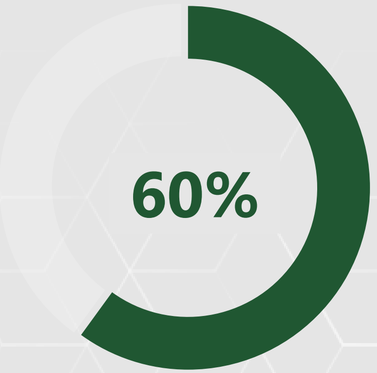


Median Years **2.3**

% OF TIME SPENT USING NODE.JS *medians*



% of Professional Development Time



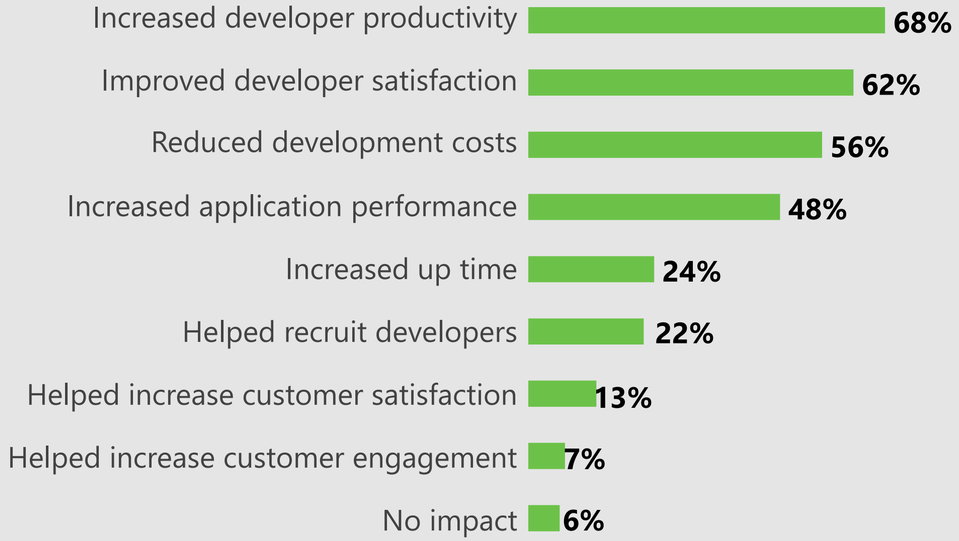
% of Total Development Time

SOURCE: Q17, Q20, Q21

Business impact

- As last year, many users say that Node.js has had a positive impact on their business – chiefly through increased developer productivity and satisfaction
- Reduced development costs and increased application performance are also important outcomes tied to Node.js
- Perhaps not surprisingly, longer-tenured users are far more likely than novices to report an impact from Node.js

HOW NODE.JS HAS IMPACTED YOUR BUSINESS



IMPACT BY YEARS USING NODE.JS *top impacts*

	< 2 yrs	2+ yrs
Increased developer productivity	58%	74%
Improved developer satisfaction	51	68
Reduced development costs	45	62
Increased application performance	42	51
Increased uptime	19	27
Helped recruit developers	15	27
No impact	9	4

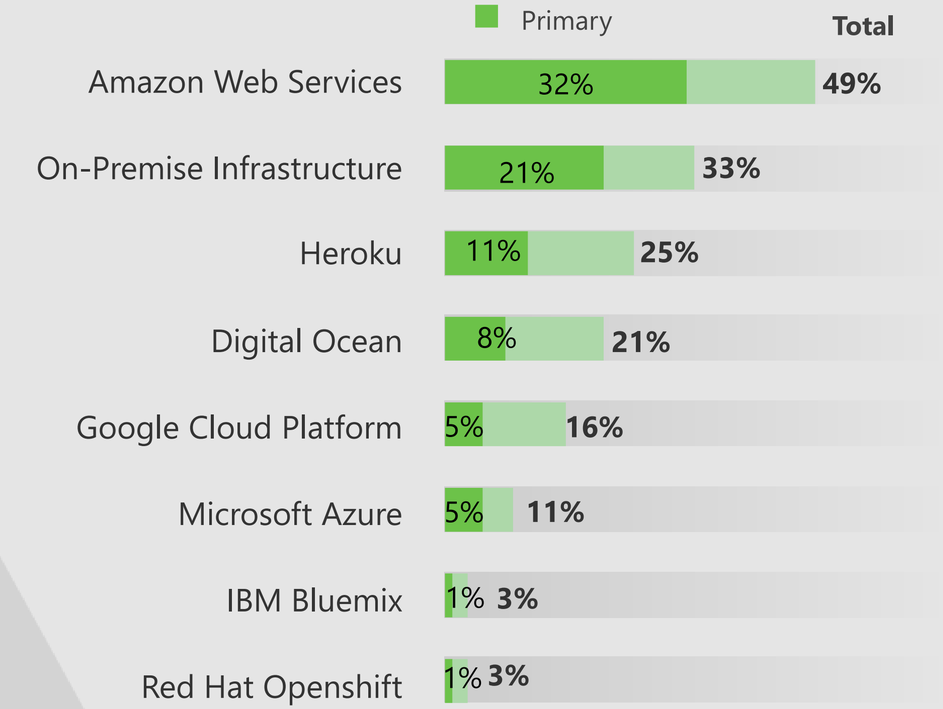
SOURCE: Q54

Where code is deployed

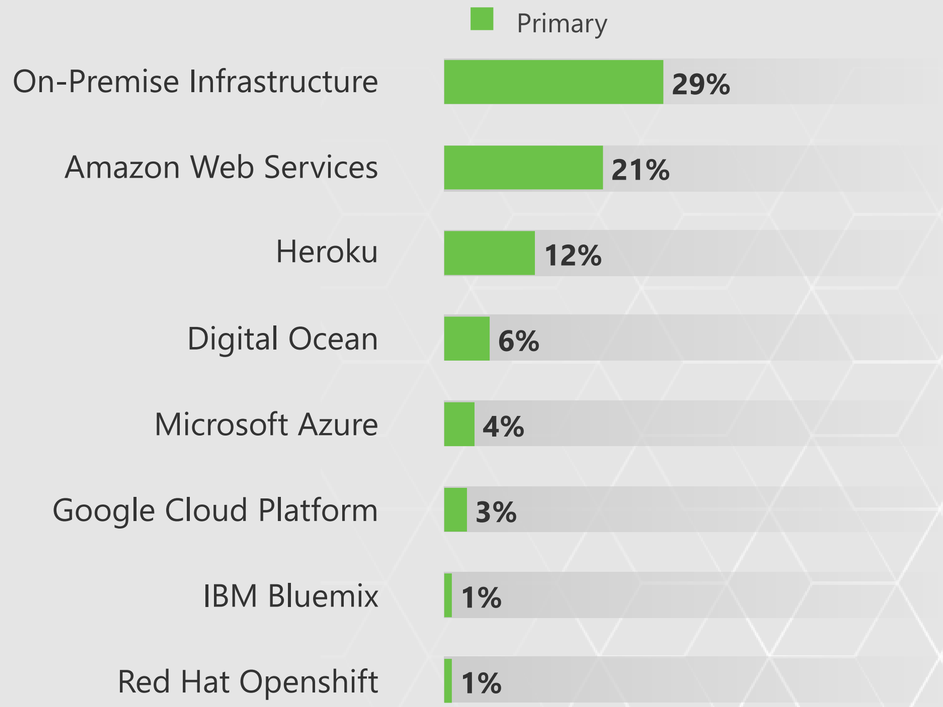
AWS is the primary place where respondents deploy code for production, and it seems to be growing for use in development

On-Premise infrastructure is also widely used, but has dropped for use in production since last year

FOR PRODUCTION



FOR DEVELOPMENT



SOURCE: Q28, Q29, Q30

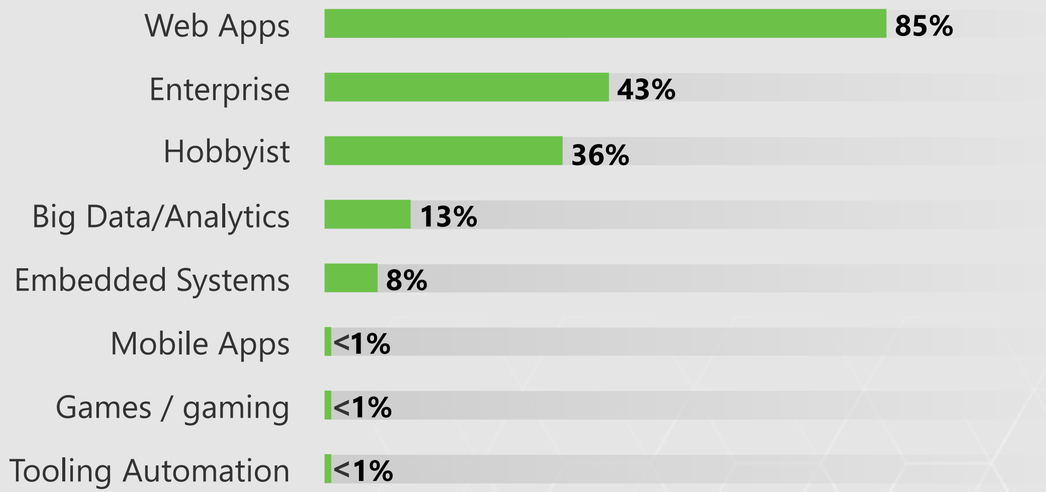
Types of development work

Majority are spending time developing web apps, particularly in full stack or front-end positions

A notable minority also engage in enterprise and/or hobbyist work

Those outside the traditional development areas are more likely to be working on embedded systems

TYPES OF DEVELOPMENT WORK SPEND TIME ON *top mentions*



TYPE OF WORK *by primary development focus*

	Back-End	Full Stack	Front-End	Other
Web Apps	**82%**	**92%**	**89%**	**66%**
Enterprise	**47**	**43**	**35**	**40**
Hobbyist	**32**	**41**	**35**	**37**
Big Data/Analytics	**13**	**14**	**7**	**13**
Embedded Systems	**8**	**7**	**3**	**19**
Hobbyist ONLY	**2**	**2**	**3**	**10**

SOURCE: Q24

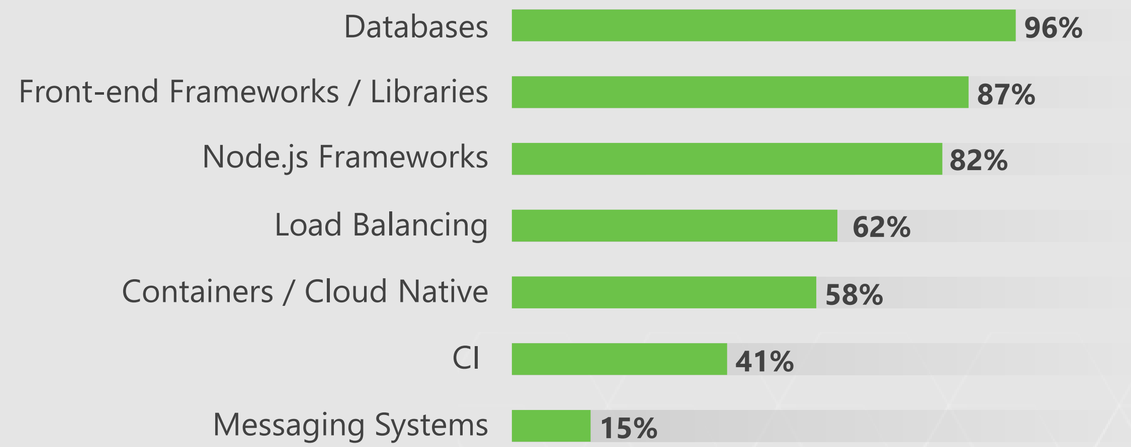
Tools / technologies used

A range of tools are used with Node.js, primarily: **databases, libraries, and Node.js frameworks**

Back-End and Full Stack developers most likely to use a range of tools

Messaging systems and CI are less commonly used than other tools – and usage has dropped since last year

TYPES OF TOOLS / TECHNOLOGIES USED WITH NODE.JS *in past 12 months*



TYPES OF TOOLS *by Primary Development Focus*

	Back-End	Full Stack	Front-End	Other
Databases	**98%**	**97%**	**89%**	**92%**
Front-end Frameworks	**83**	**93**	**94**	**69**
Node.js Frameworks	**84**	**90**	**67**	**59**
Load Balancing	**65**	**68**	**48**	**44**
Containers / Cloud	**62**	**61**	**48**	**45**
CI	**40**	**45**	**40**	**27**
Messaging	**21**	**15**	**4**	**12**

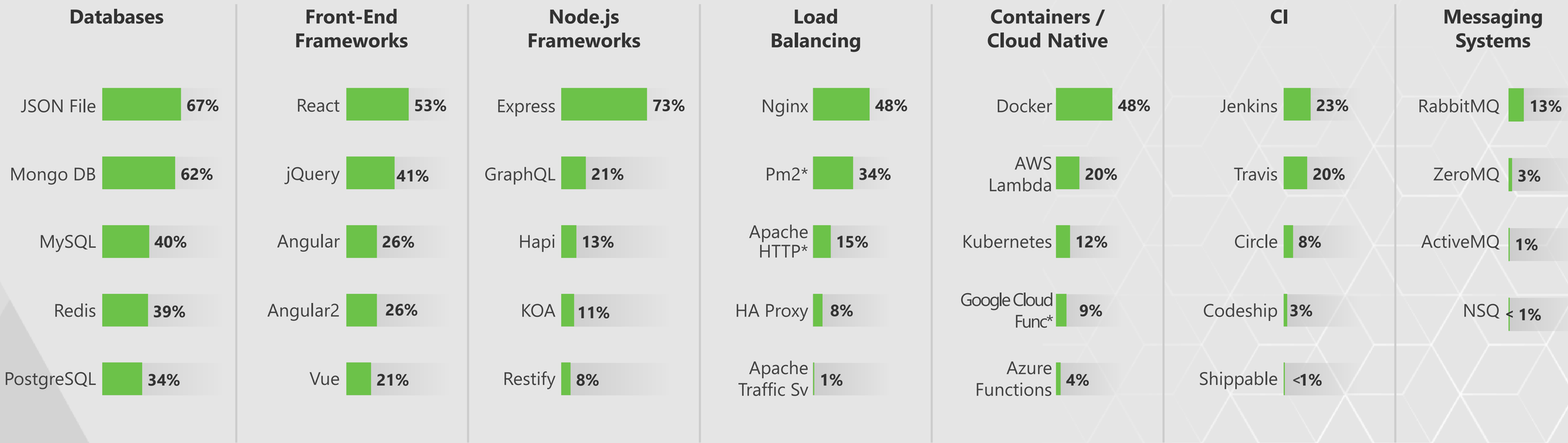
SOURCE: Q27 (net categories)

Tools / technologies used

A range of tools are used, though **one or two tools dominate within each category**

Usage of many tools has dropped while some have increased, including **Angular 2**, **Vue**, **GraphQL**, **Docker** and **Kubernetes**

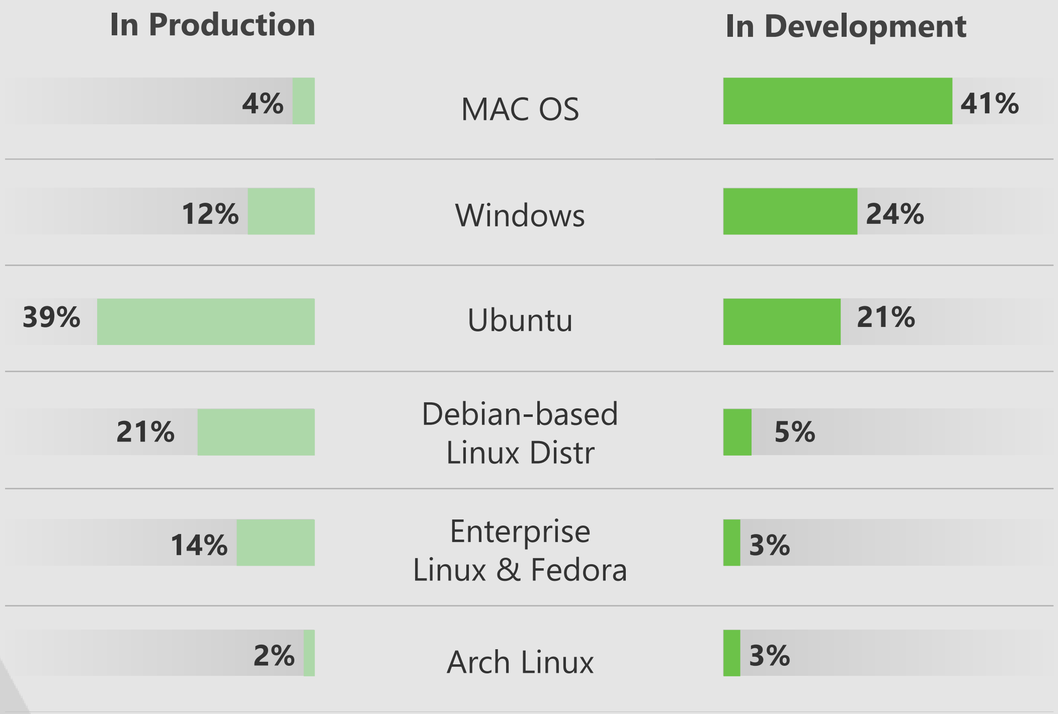
SPECIFIC TOOLS / TECHNOLOGIES USED *in past 12 months*



SOURCE: Q27

Primary OS / Distro

PRIMARY OS / DISTRO USED *top mentions*



Ubuntu is the primary OS / Distro used in production, while MAC OS is primary in development

Use of Windows – in both production and development – has increased since last year, particularly in US/CA, EMEA and in smaller companies (among other segments).

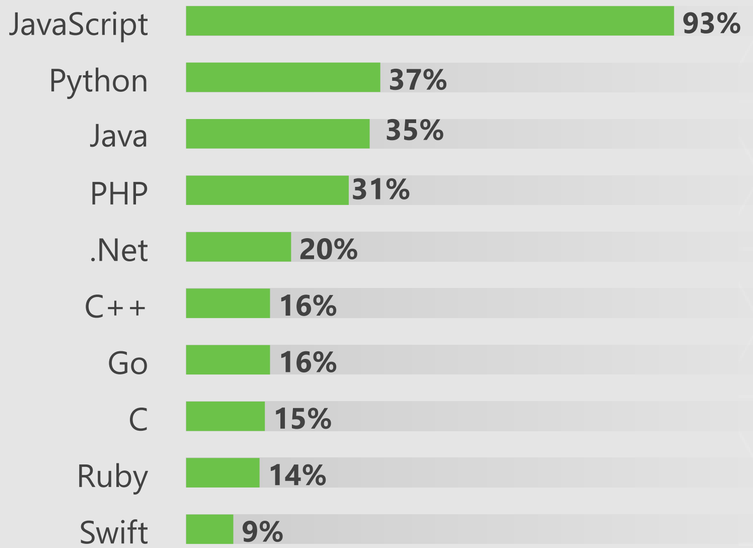
SOURCE: Q41, Q42

Languages used

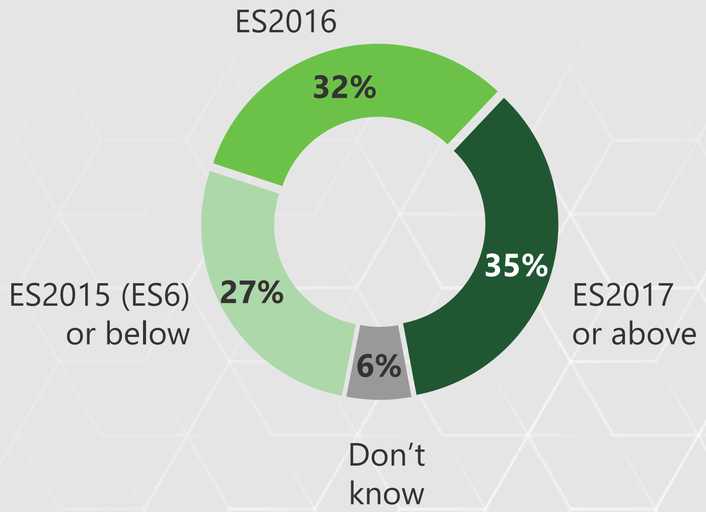
3 other languages are used on average besides Node.js – typically JavaScript and then Python, Java or PHP

More than a third are using ES2017 or above

OTHER LANGUAGES USED (PAST 12 MONTHS)



PRIMARY JAVASCRIPT LANGUAGE VERSION (6 MONTHS)

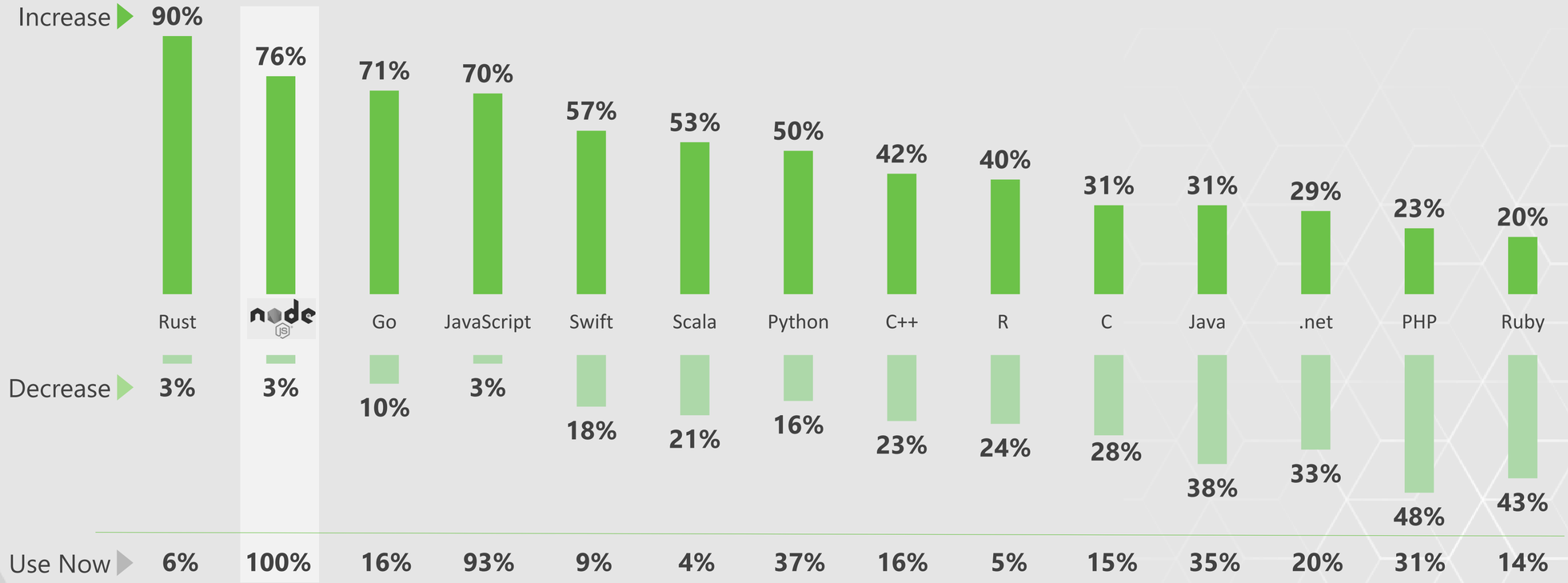


SOURCE: Q25, Q33

Expected change in other languages

- Use of other languages will also increase among current users – including Rust, Go, and JavaScript
- Usage of PHP and Ruby appears to be on the decline – although many of those in Asia/Pacific plan to increase their usage of PHP

EXPECTED CHANGE IN USE OVER NEXT 12 MONTHS *among users of each language**



SOURCE: Q25, Q26, among those who use respective brand and who provided an answer

Other languages used in addition to Node.js

Language usage varies somewhat by region and development focus

PHP is less popular in US / CA and among those outside of traditional programming areas

Despite being less likely to use JavaScript and PHP, those “other” developers use more languages on average (closer to 4), including C++ and C

BY REGION

	US / CA	Full Stack	Front-End	Other
JavaScript	**93%**	**93%**	**89%**	**96%**
Python	**39**	**35**	**35**	**39**
Java	**34**	**36**	**30**	**48**
PHP	**23**	**37**	**29**	**38**
.Net	**22**	**20**	**19**	**19**
C++	**20**	**16**	**13**	**8**
Go	**18**	**14**	**15**	**20**
C	**16**	**13**	**15**	**15**
Ruby	**20**	**12**	**8**	**11**
Average #	**3.3**	**3.2**	**2.8**	**3.3**

BY PRIMARY DEVELOPMENT FOCUS

	Back-End	Full Stack	Front-End	Other
JavaScript	**92%**	**95%**	**94%**	**84%**
Python	**36**	**38**	**33**	**41**
Java	**36**	**34**	**35**	**39**
PHP	**31**	**35**	**29**	**21**
.Net	**21**	**19**	**21**	**26**
C++	**15**	**15**	**13**	**33**
Go	**17**	**17**	**11**	**15**
C	**15**	**13**	**10**	**28**
Ruby	**12**	**14**	**14**	**16**
Average #	**3.2**	**3.2**	**2.9**	**3.6**

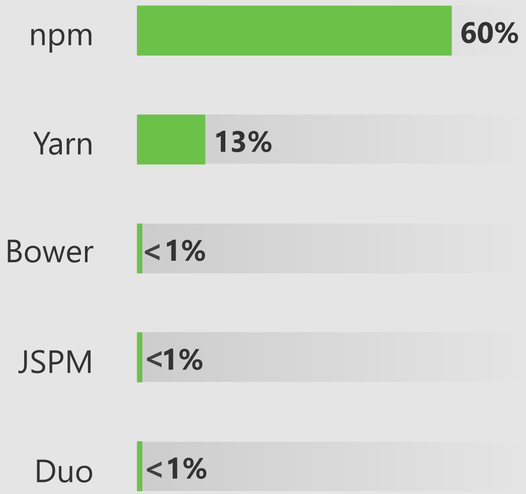
SOURCE: Q25 **Top mentions in past 12 months**

Package manager usage

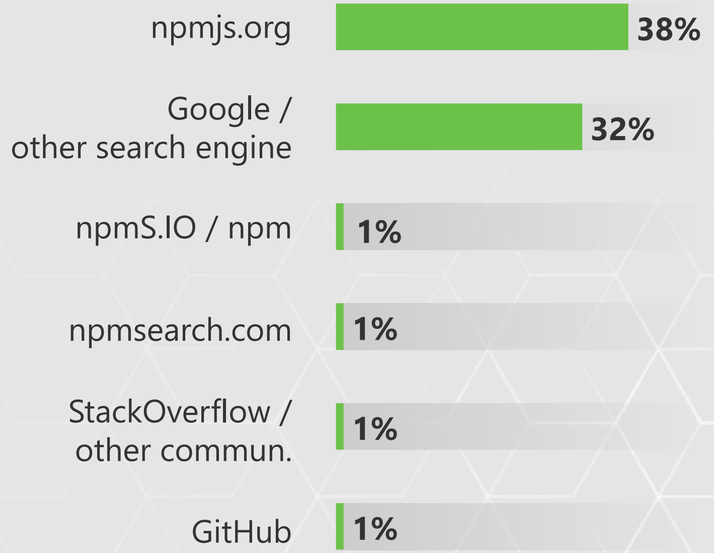
npm is, by far, the most widely used package manager – but **Yarn** is gaining in popularity in many subgroups

Respondents search for packages almost entirely on **npmjs.org** or through **Google / search engine**, which are gaining popularity in APAC and among other developers

PACKAGE MANAGER USAGE



WHERE PACKAGES ARE SEARCHED FOR



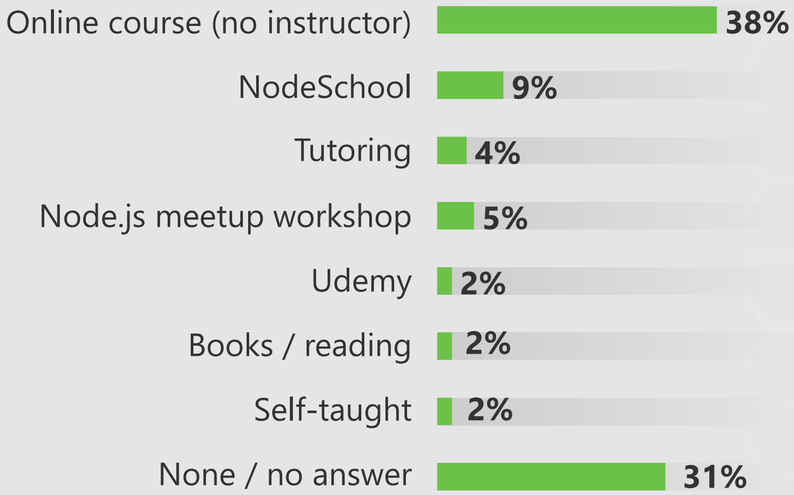
SOURCE: Q47/Q48 (including blank/no answer)

Informal coding education

The primary way respondents are gaining informal coding education is via online courses without an instructor – particularly outside US/CA.

While not widely used, tutoring is more popular in APAC than in other regions.

INFORMAL CODING EDUCATION



OTHER SOURCES *mentioned by 1% or fewer*

- Google / YouTube
- NodeBots
- Code School
- Code Academy
- Coding Bootcamp
- Online tutorials
- NodeTogether
- NodeGirls
- Node.js documentation
- PluralSight
- github
- Freecode Camp
- Blogs/articles

SOURCE: Q18

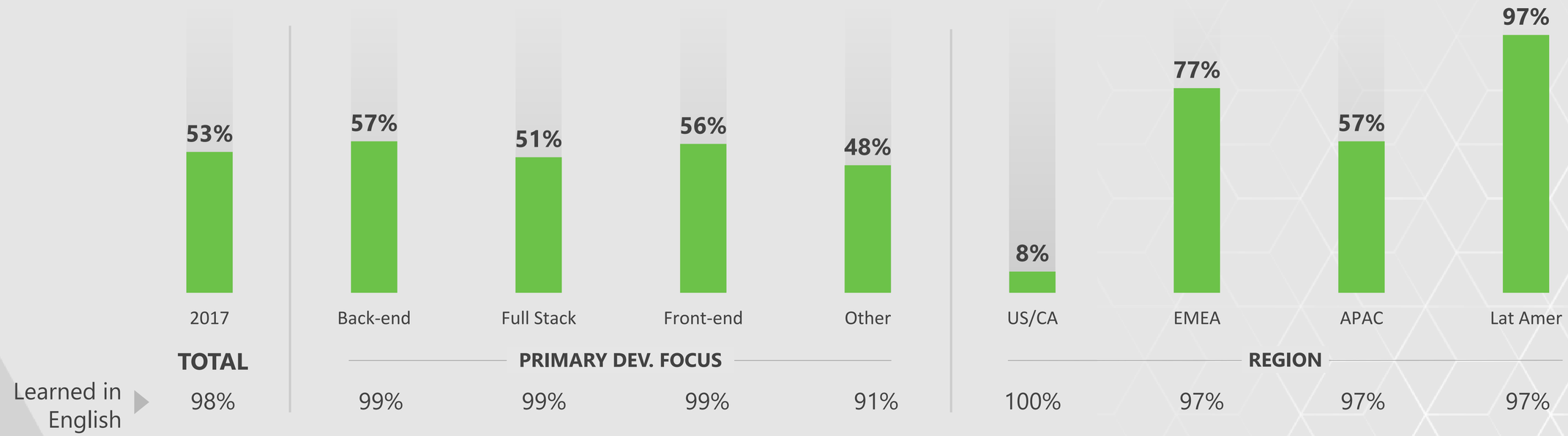
***Single response question, but allowed multiple responses in 'other specify' in 2017; not trendable**

Node.js Skills Acquisition

Nearly all respondents learned Node.js in English – which is non-native for more than half of respondents

In certain regions – including EMEA and (especially) Latin America – a vast majority of respondents learned Node.js in a non-native language

LEARNED NODE.JS IN NON-NATIVE LANGUAGE



SOURCE: Q7/Q8

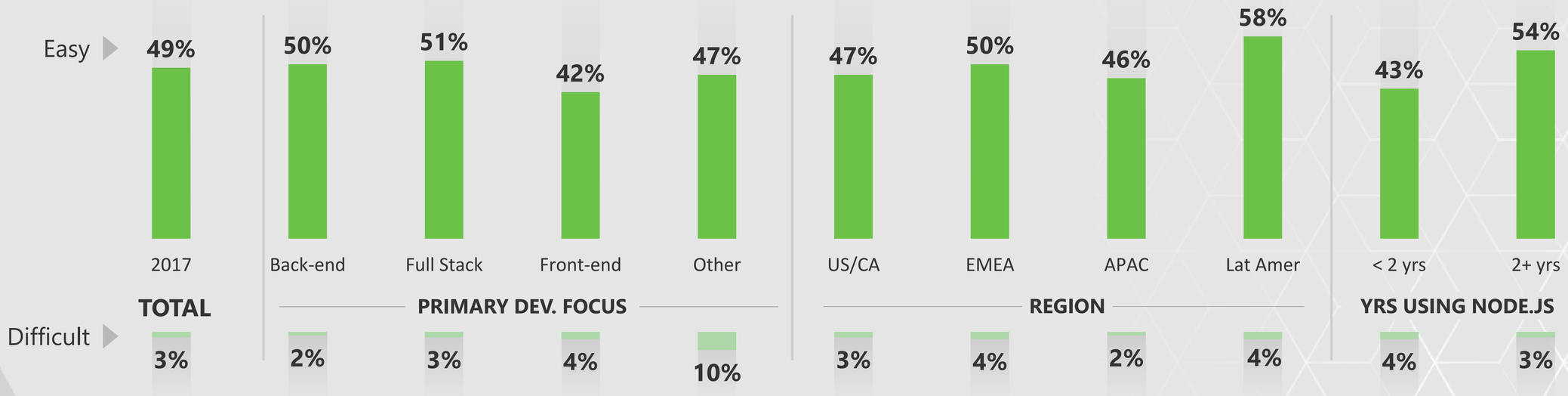
Ease of learning Node.js

About **half of respondents say it was generally easy** to learn. Very few complain that it is difficult

Surprisingly, those in Latin America – who are most likely to have learned in a non-native language – are particularly upbeat about ease of learning

Newer users are less enthusiastic than those using it 2+ years – suggesting that **more could be done to improve the learning experience**

EASE OF LEARNING NODE.JS *by subgroup*



SOURCE: Q34, among those with an opinion; blanks excluded. *** Top 3/Bottom 3 box on scale from 1 'extremely difficult' to 10 'extremely easy'.**

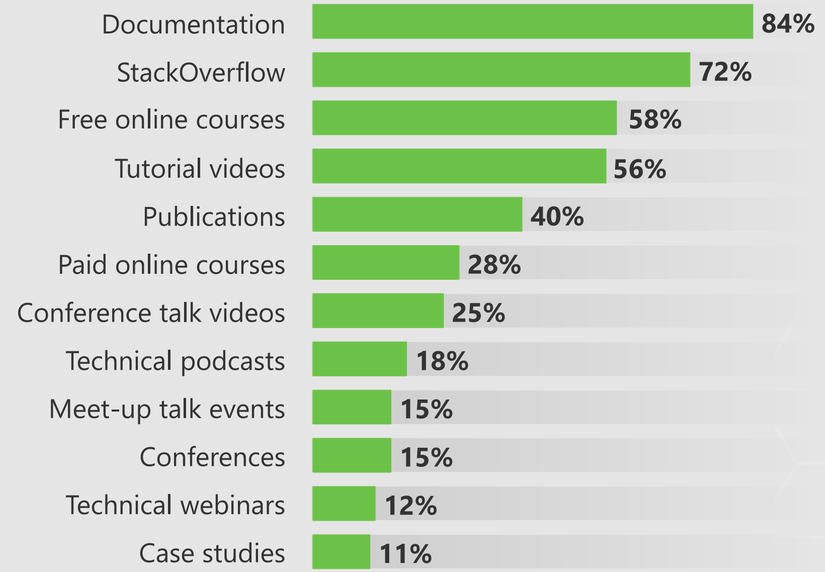
Resources used

Respondents use many resources when learning a new language – documentation and StackOverflow chief among them

Tutorial videos are also widely used

While documentation and StackOverflow are still top, newer Node.js users are more likely than others to also use free & paid online courses and tutorial videos

RESOURCES RELY ON MOST WHEN LEARNING NEW LANGUAGE / FRAMEWORK / RUNTIME ENVIRONMENT



RESOURCES USED BY YEARS *using Node.js*

	< 2 yrs	2+ yrs
Documentation	**80%**	**88%**
StackOverflow	**71**	**72**
Free online courses	**66**	**52**
Tutorial videos	**62**	**52**
Publications	**39**	**40**
Paid online courses	**35**	**24**
Conference talk videos	**19**	**29**
Technical podcasts	**16**	**19**
Meetup talk events	**10**	**17**
Conferences	**11**	**18**

OTHER RESOURCES *used by 8% or less*

Meet-up coding events
Workshops (independ-ent of conf's)
Enterprise tooling
Enterprise services
Biz case study podcasts
Biz case study webinars

SOURCE: Q37, among those who provided a response (no answer excluded) ***Not trended by subgroup**

Methodology

- This report presents selected findings from the 2018 Node.js User Survey.
- The primary objective of the research was to profile Node.js users and identify potential areas of improvement. The findings will be used for program development, marketing and PR/external communications.
- The study was conducted online from Oct 5, 2017 to January 7, 2018 via a self-administered survey.
- The survey was fielded worldwide in English and Chinese to encourage maximum response.
- The survey link was distributed by the Node.js Foundation through a number of channels including email, Twitter, conferences, blogs and word of mouth (meet-ups).
- A total of **1,626** individuals responded to at least some questions in the survey.*
- To ensure data integrity and unbiased interpretation, data analysis and reporting was conducted by Research Collaborative, an independent market research firm.
- Numbers may not total to 100% due to rounding

*** Due to skip logic or missed questions, the number of responses for individual questions may be considerably lower**

Thank you

For more information:

Contact [presenter]

+1 (xxx) xxx-xxxx

xxxxxxxx@yyyyy.zzz

2019 NODE.JS USER STUDY DETAILED REPORT OF FINDINGS

May 2018

Appendix

THE LINUX FOUNDATION

Overview / methodology

- This report presents selected findings from the 2018 Node.js User Survey.
- The primary objective of the research was to profile Node.js users and identify potential areas of improvement. The findings will be used for program development, marketing and PR/external communications.
- The study was conducted online from Oct 5, 2017 to January 7, 2018 via a self-administered survey.
- The survey was fielded worldwide in English and Chinese to encourage maximum response.
- The survey link was distributed by the Node.js Foundation through a number of channels including email, Twitter, conferences, blogs and word of mouth (meet-ups).
- A total of **1,626** individuals responded to at least some questions in the survey.* ^
- To ensure data integrity and unbiased interpretation, data analysis and reporting was conducted by Research Collaborative, an independent market research firm.
- Numbers may not total to 100% due to rounding

*** Due to skip logic or missed questions, the number of responses for individual questions may be considerably lower. ^ For a population of 10M, a sample size of 384 is needed for a confidence level of 95% and confidence interval of +/- %5. See https://www.surveysystem.com/sscalc.htm**

SECTION HIGHLIGHTS

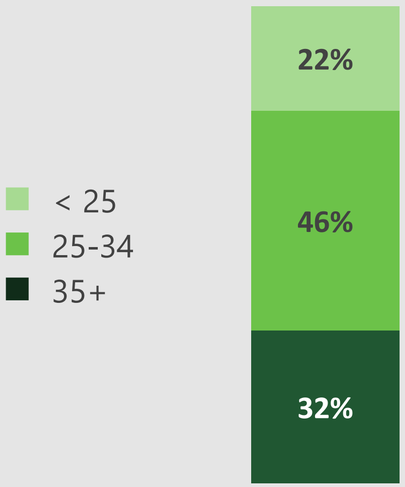
Business / personal profile

- As with last year, the typical Node.js user is male, age 31 and college educated.
- A majority are developers, in small (<100 employees) companies, with 5+ years of professional development experience.
- Although many have 10+ years total development experience, this year's respondents are less experienced in terms of total development experience.
- Respondents come from across the globe, but most are in US/CA or EMEA.
- Collectively, respondents speak over 60 languages, but for nearly half English is primary.
- The mix of countries has changed somewhat since last year – with fewer from US and China and more from India and Canada.
- There are considerable differences in personal and business characteristics, with those from US/CA older, more experienced, and from larger companies than users elsewhere.

Personal characteristics

- The typical Node.js user is male, age 31, college educated and white.
- Respondents are slightly older in this wave, but gender and education are unchanged.

RESPONDENT AGE



Median Age: **31.0**

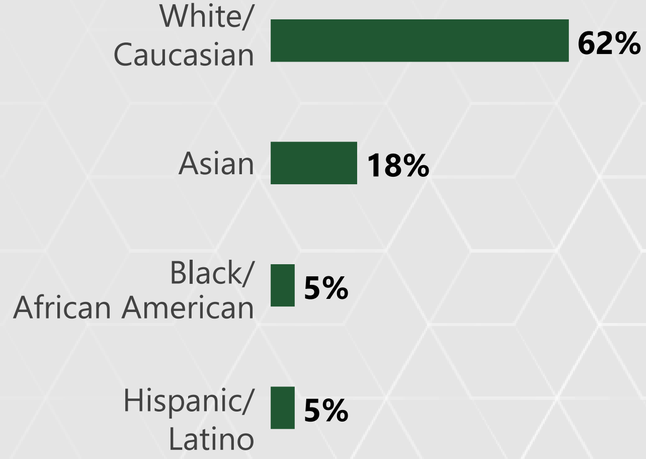
GENDER

Male	**94%**
Female	**5%**
Other	**1%**

EDUCATION

High school or less	**16%**
Some college	**10%**
College degree	**51%**
Grad degree	**21%**
Other	**2%**

ETHNICITY*



SOURCE: Q9, Q10, Q11, Q14

*** Ethnicity asked as open end this year, closed end last year; not trendable**

Profile by region

- There are notable differences in business and personal profile by region.
- Those in the US/CA are older, more experienced and work in larger companies than their peers around the globe.
- EMEA respondents are particularly highly educated.
- The profile of APAC respondents has changed in many respects vs. last wave – perhaps not surprising given the drop in China as a percentage of APAC respondents this year. APAC respondents are relatively new to development.

BUSINESS CHARACTERISTICS *by region*

	US/CA	EMEA	APAC	LatAm
Co size (median # ee's)	**99**	**24**	**24**	**40**
Prof'l dev experience (median yrs)	**7.5**	**5.3**	**3.7**	**4.4**
Prof'l dev experience (10+ yrs)	**42%**	**29%**	**18%**	**28%**
Developers	**60%**	**67%**	**64%**	**70%**
Managers	**29%**	**23%**	**24%**	**24%**

PERSONAL CHARACTERISTICS *by region*

	US/CA	EMEA	APAC	LatAm
English primary language	**92%**	**21%**	**42%**	**1%**
Age (median)	**33**	**31**	**29**	**30**
Age (% 35+)	**41%**	**31%**	**20%**	**24%**
Male	**91%**	**95%**	**97%**	**96%**
Have grad degree	**15%**	**28%**	**20%**	**8%**

SOURCE: Q1, Q3, Q7, Q9, Q14. Q15

Node.js usage profile

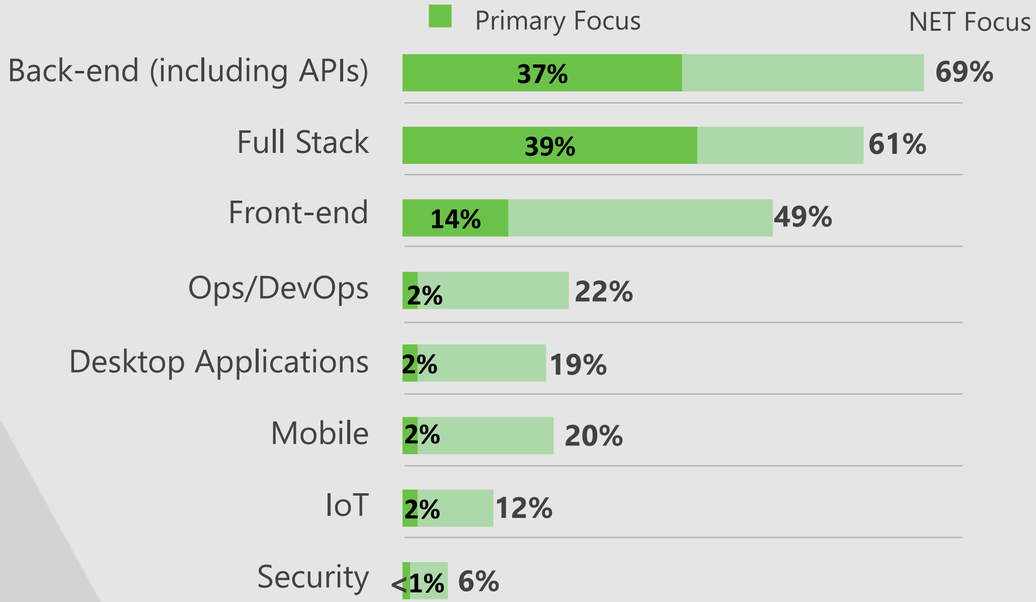
- As with last year, the typical user has been working with Node.js over 2 years, and spends more than half of his development time using it. The vast majority are developing web apps.
- Most respondents are primarily back-end or full stack development-focused, and they use Node.js more regularly at work than others do.
- Relatively few users are primarily focused on “other” non-traditional development areas*, but those who are tend to be older and more highly educated than others.
- Users are deploying through a range of channels, but AWS is most widely used for production and On-premise or AWS for development. Heroku seems to be growing in popularity in APAC and Latin America.
- More than 4 in 5 back-end and full stack developers are using Node.js frameworks; Express is tops, but Graph QL is increasingly prevalent this wave.
- Most are using a transpiler and module bundler (especially full stack and front-end developers). Babel is the preferred transpiler, but Typescript is growing. Webpack continues to dominate the module bundler space.
- Ubuntu is the primary OS/Distro used in production, and MAC OS in development – but Windows seems to be growing in popularity for both (especially in US/CA and EMEA).

*Including DevOps, Desktop Apps, Mobile, IoT and Security

Development focus

- Three in four Node.js users are focused primarily on back-end or full stack development
- There has been a slight drop in this wave in those who have any focus on back-end development
- US / CA respondents are more focused on full stack than back-end
- Ops / DevOps is really only a focus in US / CA – although even here, it is not widely focused on

DEVELOPMENT FOCUS *title*



PRIMARY DEVELOPMENT FOCUS *by region*

Primary Development Focus by Region

	US/CA	EMEA	APAC	LatAm
Back-end	**32%**	**38%**	**43%**	**42%**
Full Stack	**42**	**39**	**38**	**41**
Front-end	**15**	**14**	**13**	**11**
Ops / DevOps	**3**	**1**	**<1**	**1**
Desktop Apps	**2**	**2**	**2**	**1**
Mobile	**2**	**2**	**2**	**2**
IoT	**1**	**2**	**1**	**2**
Security	**<1**	**1**	**<1**	**0**

SOURCE: Q22, Q23

Profile by development area

- There are some differences in users' business and personal characteristics based on development focus.
- Full stack developers have been using Node.js the longest, and along with back-end developers, spend the most time with it.
- Those outside of the three traditional development areas tend to be older and more highly educated than others.

BUSINESS CHARACTERISTICS *by primary development focus*

	Back-end	Full Stack	Front-end	Other
Co size (median # ee's)	**54**	**37**	**79**	**44**
Prof'l dev experience (median yrs)	**4.7**	**6.0**	**5.6**	**6.1**
Prof'l dev experience (10+ yrs)	**31**	**32**	**29**	**38**
Developers	**69%**	**61%**	**71%**	**50%**
Managers	**25%**	**28%**	**19%**	**32%**
Years using Node.js (median)	**2.2**	**2.5**	**2.3**	**2.2**
% Prof Dev time Use Node.js (median)	**61%**	**62%**	**27%**	**39%**

PERSONAL CHARACTERISTICS *by primary development focus*

	Back-end	Full Stack	Front-end	Other
English primary language	**43%**	**48%**	**44%**	**48%**
Age (median)	**31**	**31**	**31**	**35**
Male	**94%**	**95%**	**95%**	**91%**
Have grad degree	**22%**	**19%**	**18%**	**29%**
US/CA	**27%**	**34%**	**35%**	**39%**
EMEA	**46%**	**43%**	**45%**	**45%**
APAC	**19%**	**16%**	**15%**	**10%**
Latin America	**8%**	**8%**	**6%**	**4%**

"Other" includes: Ops/Dev Ops, Desktop Applications, Mobile, IoT and Security

SOURCE: Q1, Q3, Q4, Q7, Q9, Q10, Q15, Q17, Q20

Where deploy code

- EMEA respondents are less likely than others to use AWS, preferring on-premise infrastructure. US/CA respondents are also likely to be deploying via on-premise infrastructure
- Heroku is growing in both APAC and Latin America, and is one of the top choices for deployment for development in Latin America

WHERE PRIMARILY DEPLOY NODE.JS CODE *by region*

For Production

	US/CA	EMEA	APAC	LatAm
Amazon Web Services	**37%**	**23%**	**42%**	**40%**
On-Premise Infrastructure	**20**	**27**	**11**	**12**
Heroku	**11**	**11**	**12**	**16**
Digital Ocean*	**7**	**9**	**7**	**7**
Google Cloud	**4**	**5**	**5**	**6**
Microsoft Azure^	**5**	**4**	**7**	**5**
IBM Bluemix	**2**	**1**	**1**	**3**
Red Hat Openshift^	**1**	**1**	**2**	**1**
Depl not req'd	**5**	**7**	**4**	**4**

For Development

	US/CA	EMEA	APAC	LatAm
On-Premise Infrastructure	**32%**	**32%**	**20%**	**22%**
Amazon Web Services	**24**	**17**	**27**	**20**
Heroku	**9**	**12**	**16**	**20**
Digital Ocean*	**5**	**6**	**7**	**5**
Google Cloud	**3**	**4**	**4**	**1**
Microsoft Azure^	**3**	**4**	**7**	**4**
IBM Bluemix	**1**	**1**	**1**	**4**
Red Hat Openshift^	**1**	**1**	**1**	**1**
Depl not req'd	**16**	**16**	**14**	**13**

SOURCE: Q28, Q30

^ Attribute wording change; interpret trend cautiously

*** Items added to pre-list in 2017; not trended**

Where deploy code

- Deployment also varies somewhat based on development focus.
- AWS is widely used by back-end, full stack and front-end developers, but less so for others.
- Heroku is relatively popular among full-stack developers.

WHERE PRIMARILY DEPLOY NODE.JS CODE *by primary development focus*

For Production

	Back-end	Full Stack	Front-end	Other
Amazon Web Services	**36%**	**33%**	**29%**	**16%**
On-Premise Infrastructure	**23**	**20**	**24**	**20**
Heroku	**10**	**14**	**9**	**8**
Digital Ocean*	**5**	**11**	**8**	**5**
Google Cloud	**5**	**5**	**4**	**5**
Microsoft Azure^	**5**	**4**	**4**	**9**
IBM Bluemix	**2**	**1**	**0**	**3**
Red Hat Openshift^	**1**	**1**	**1**	**2**
Depl not req'd	**4**	**3**	**13**	**13**

For Development

	Back-end	Full Stack	Front-end	Other
On-Premise Infrastructure	**29%**	**29%**	**33%**	**25%**
Amazon Web Services	**24**	**22**	**17**	**10**
Heroku	**12**	**14**	**9**	**7**
Digital Ocean*	**3**	**8**	**7**	**4**
Google Cloud	**3**	**4**	**2**	**4**
Microsoft Azure^	**5**	**3**	**2**	**7**
IBM Bluemix	**1**	**1**	**1**	**2**
Red Hat Openshift^	**1**	**1**	**1**	**2**
Depl not req'd	**15**	**12**	**20**	**27**

SOURCE: Q28, Q30

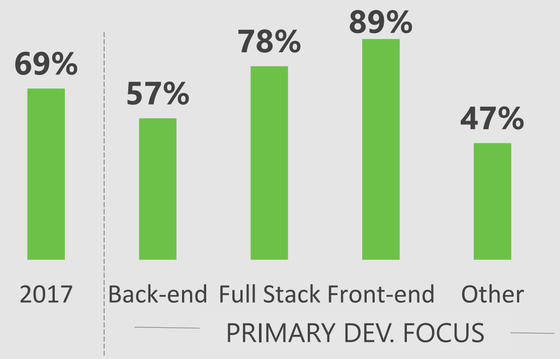
^ Attribute wording change; interpret trend cautiously

*** Items added to pre-list in 2017; not trended**

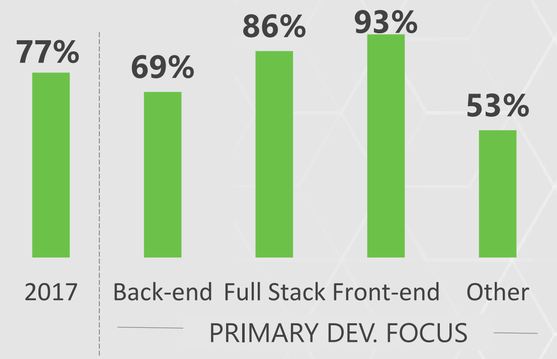
Transpilers and bundlers

- Most respondents use a transpiler and module bundler – particularly front-end and full stack developers.
- Use of transpilers has risen since last year; Babel is most common, but use of Typescript is on the rise.
- Webpack seems to be solidifying its notable lead among module bundlers.

USE A TRANSPILER



USE A MODULE BUNDLER



SOURCE: Q31, Q32

Primary OS / distro

- Distro use varies somewhat by region, with Ubuntu more popular in APAC and Latin America, and MAC OS more popular in US / CA
- Debian-based Linux, while not widely used anywhere for development, is somewhat popular in EMEA and Latin America in production

PRIMARY OS / DISTRO USED *by region*

In Production					In Development			
US/CA	EMEA	APAC	LatAm		US/CA	EMEA	APAC	LatAm
6%	**3%**	**3%**	**3%**	MAC OS	**50%**	**35%**	**40%**	**33%**
12	**12**	**14**	**3**	Windows	**22**	**27**	**23**	**15**
37	**37**	**45**	**41**	Ubuntu	**15**	**21**	**27**	**37**
14	**26**	**16**	**28**	Debian-based Linux	**4**	**6**	**4**	**5**
16	**11**	**15**	**17**	Ent. Linux & Fedora	**4**	**3**	**3**	**3**
3	**3**	**2**	**1**	Arch Linux	**2**	**4**	**3**	**4**

SOURCE: Q41, Q42

Primary OS / distro

- Primary distro varies somewhat by development focus
- Ubuntu is most popular among back-end and full stack developers, while Windows is more popular among front-end and “other” developers (where it is the #1 choice for both production and development)

PRIMARY OS / DISTRO USED *by primary development focus*

In Production					In Development			
Back-End	Full Stack	Front-End	Other		Back-End	Full Stack	Front-End	Other
3%	**3%**	**8%**	**6%**	MAC OS	**36%**	**47%**	**48%**	**26%**
8	**9**	**18**	**29**	Windows	**21**	**21**	**29**	**39**
42	**41**	**30**	**25**	Ubuntu	**26**	**20**	**14**	**20**
19	**24**	**19**	**15**	Debian-based Linux	**5**	**5**	**2**	**6**
17	**12**	**14**	**9**	Ent. Linux & Fedora	**6**	**3**	**1**	**0**
3	**2**	**3**	**2**	Arch Linux	**4**	**2**	**4**	**3**

SOURCE: Q41, Q42

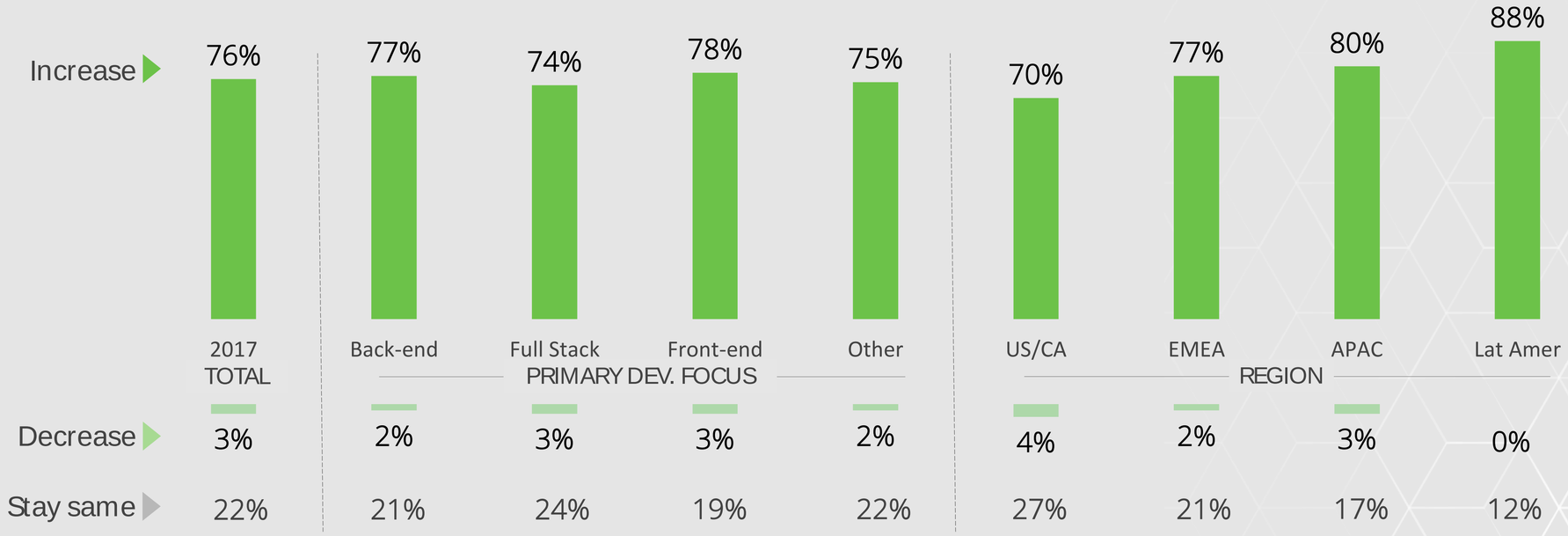
Languages Used

- Node.js users are using a range of other languages besides Node.js – more than 3 on average, including primarily JavaScript, Python, Java and PHP.
- A third are using ES2017 or above – three times as many as in last year.
- APAC users use fewer languages on average than others, while those outside traditional development areas use more – particularly C++ and C.
- Most expect to increase their use of Node.js over the next 12 months – and the number is rising. Growth will likely come from outside the US/CA – particularly in Latin America or EMEA.
- Use of other languages is also expected to increase – including Rust, Go and JavaScript.
- Usage of Ruby has dropped, and users are far more likely to say they will "decrease" usage than increase over the next 12 months.
- PHP is less popular in US/CA and among "other" developers; and, many of those who use it say they will decrease usage over the next 12 months.
- Go and Swift may be stealing the attention of Node.js users – many of those who plan to maintain/decrease with Node.js will increase with Go or Swift in the next 12 months.

Expected change in Node.js usage

- Three quarters of respondents say they plan to increase their use of Node over the next 12 months – up from 2016
- The rise in increased usage is attributable to back-end developers and those in EMEA, although Latin American respondents are most likely to say they will increase their usage

EXPECTED CHANGE IN USE OF NODE.JS OVER NEXT 12 MONTHS *by subgroup*

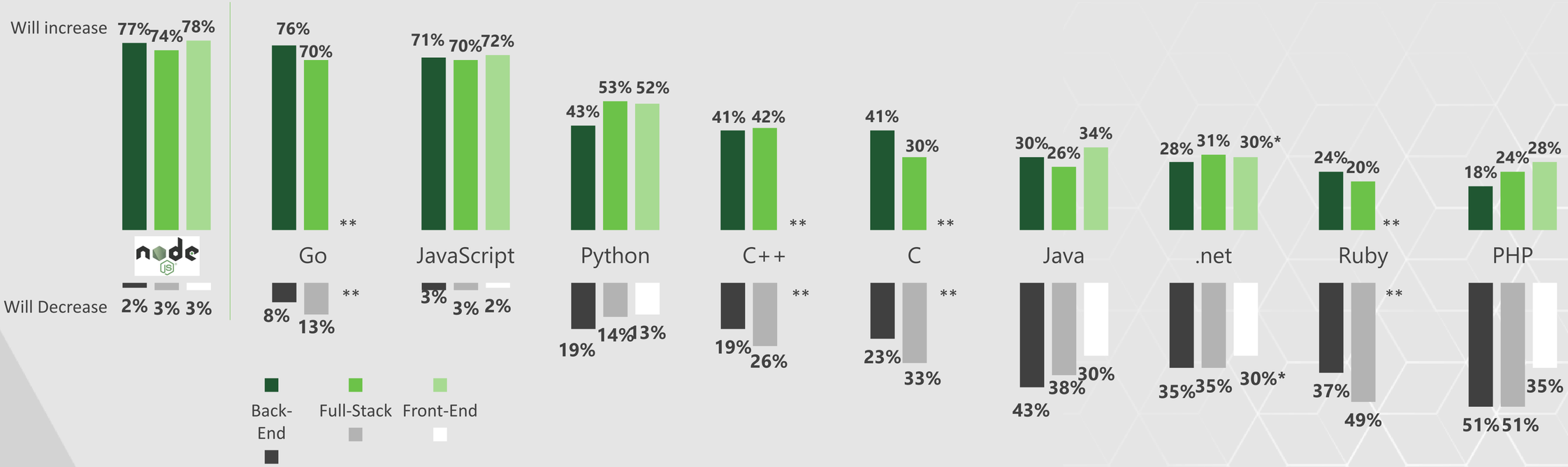


SOURCE: Q26, among Node.js users who provided an answer

Expected change in other languages

- Planned usage does not vary considerably across the three main developer segments – except that back-end developers are particularly likely to decrease their use of Java and front-end developers are less likely to decrease their use of PHP.

EXPECTED CHANGE IN USE OVER NEXT 12 MONTHS *among users of each language**



SOURCE: Q25, Q26, among those who use respective brand and who provided an answer

*** Sample size small (n<50)**

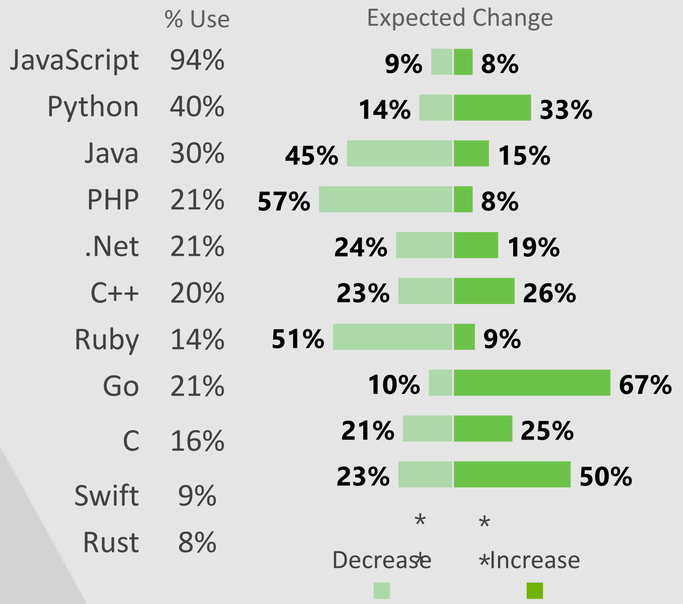
****Sample size too small to report (n<30)**

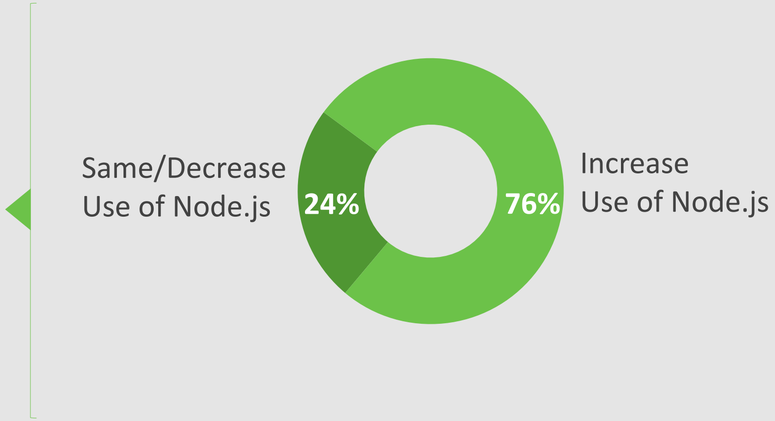
Expected change in other languages

- Those who plan to increase their use of Node.js will be increasing their use of a number of other languages as well
- Many of those who will hold steady/decrease with Node.js will be increasing their focus on Go or Swift

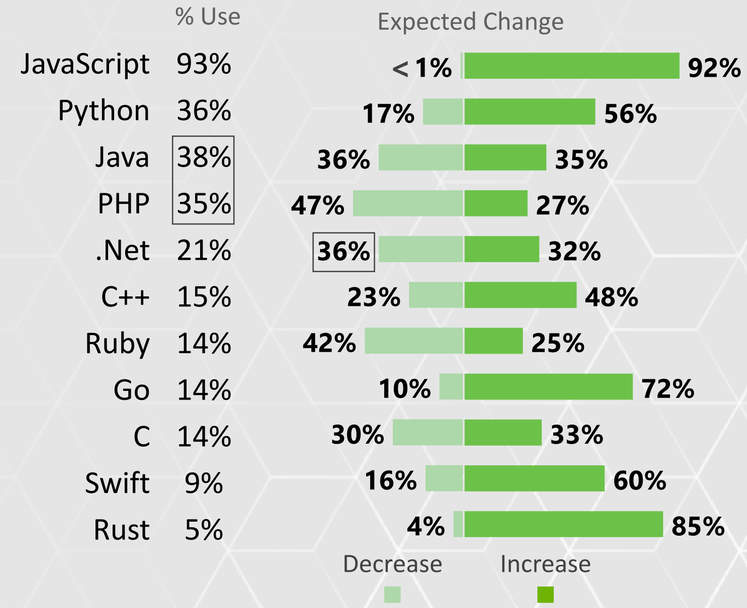
EXPECTED CHANGE IN USE OVER NEXT 12 MONTHS

OTHER LANGUAGES: STEAD/DECREASE NODE.JS





OTHER LANGUAGES: INCREASE NODE.JS

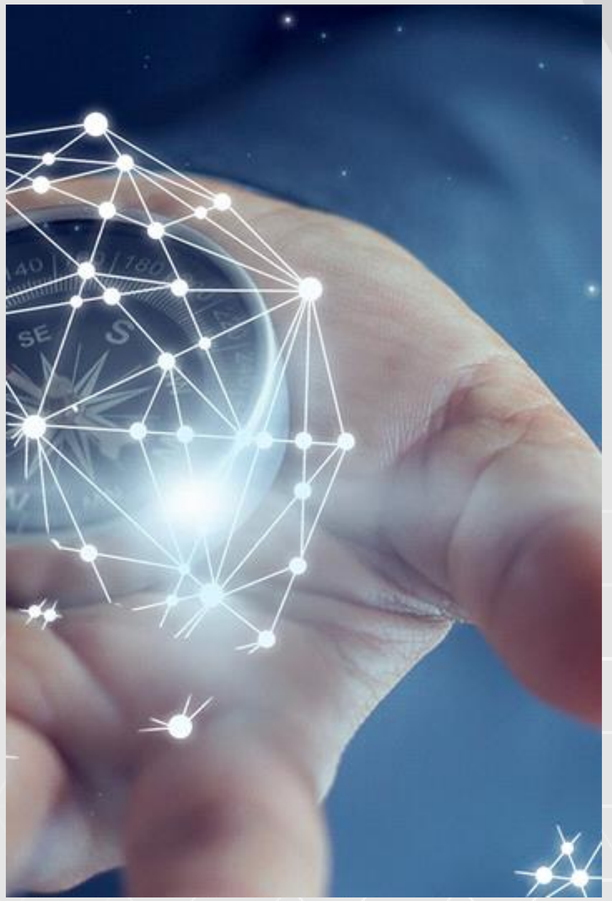


SOURCE: Q25, Q26, among those who use respective brand and who provided an answer

SECTION HIGHLIGHTS

Package Managers

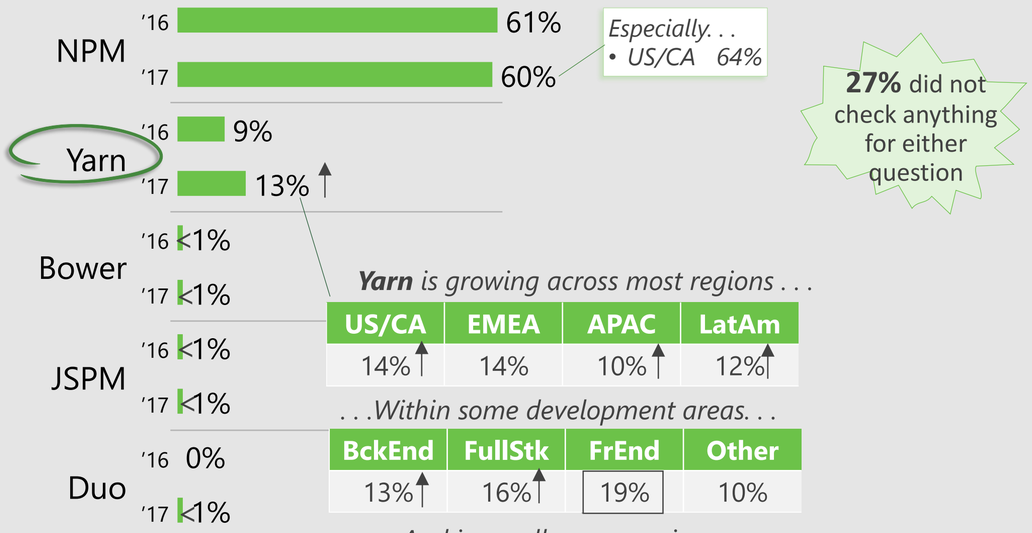
- npm is by far the most widely used package manager, but Yarn is gaining in popularity.
- Node users search for packages primarily on ptmjs.org or Google/search engines; the use of Google/search engines has increased since last year.
- It is becoming increasingly important to users to manage different packages for multi environments. Those in APAC and Latin American regions are most likely to see this as a priority.
- Availability of multiple registries is not widely seen as important in certain segments like EMEA, US/CA and small companies.
- Latin America is the only area where having multiple registries *is* an important priority.



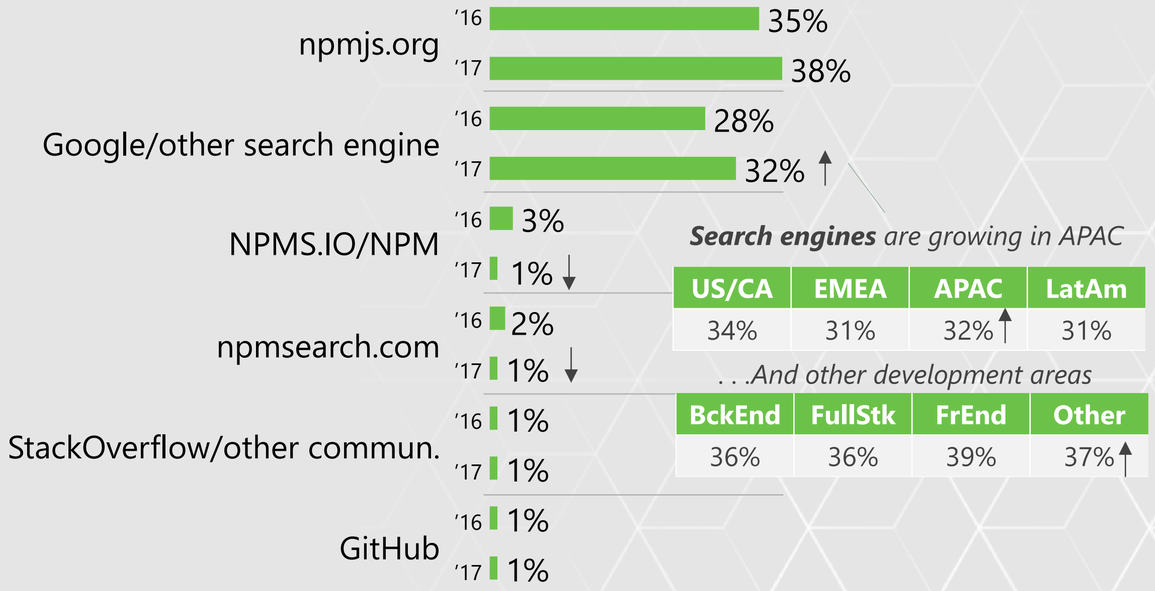
Package Manager Usage

- NPM is, by far, the most widely used package manager – but Yarn is gaining in popularity in many subgroups.
- Respondents search for packages almost entirely on npmjs.org or through Google/search engine, which are gaining popularity in APAC and among other developers.

Package Manager Using



Where Search For Packages

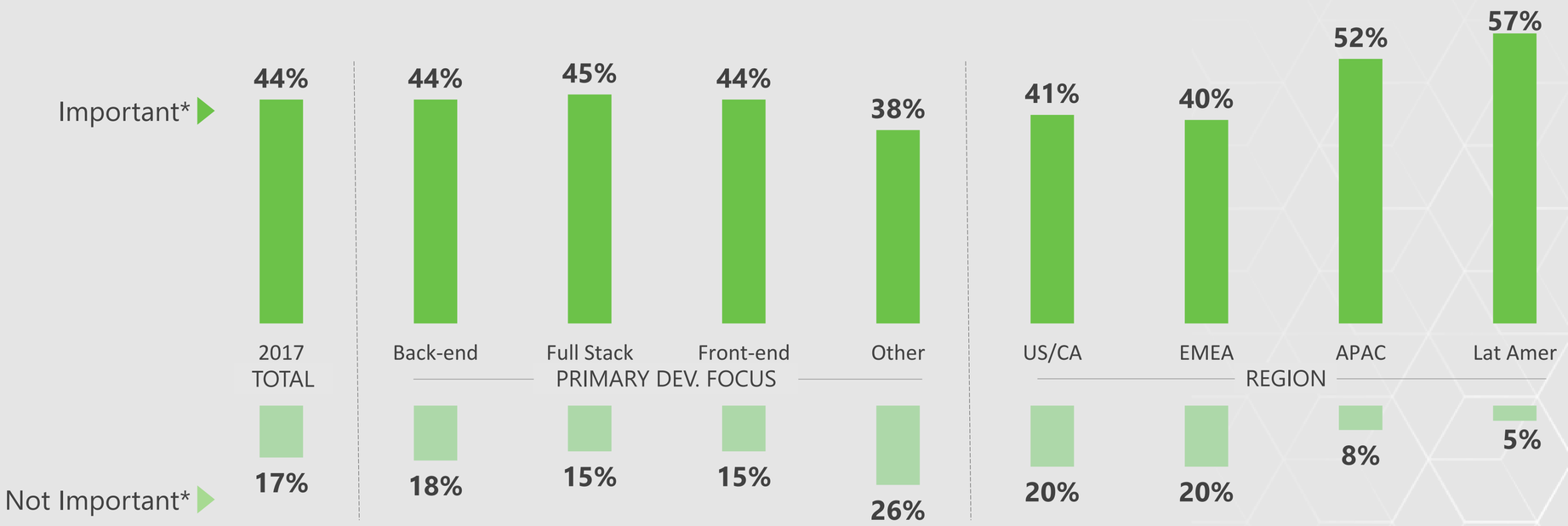


SOURCE: Q47/Q48 (including blank/no answer)

Managing different packages

- It is becoming increasingly important to be able to manage different packages for multi environments – the rise most evident among full stack developers and those in US / CA
- These rises notwithstanding, managing different packages is particularly important to those in APAC and Latin America

IMPORTANCE OF MANAGING DIFFERENT PACKAGES FOR MULTI ENVIRONMENTS *e.g., react vs react-native or cli*

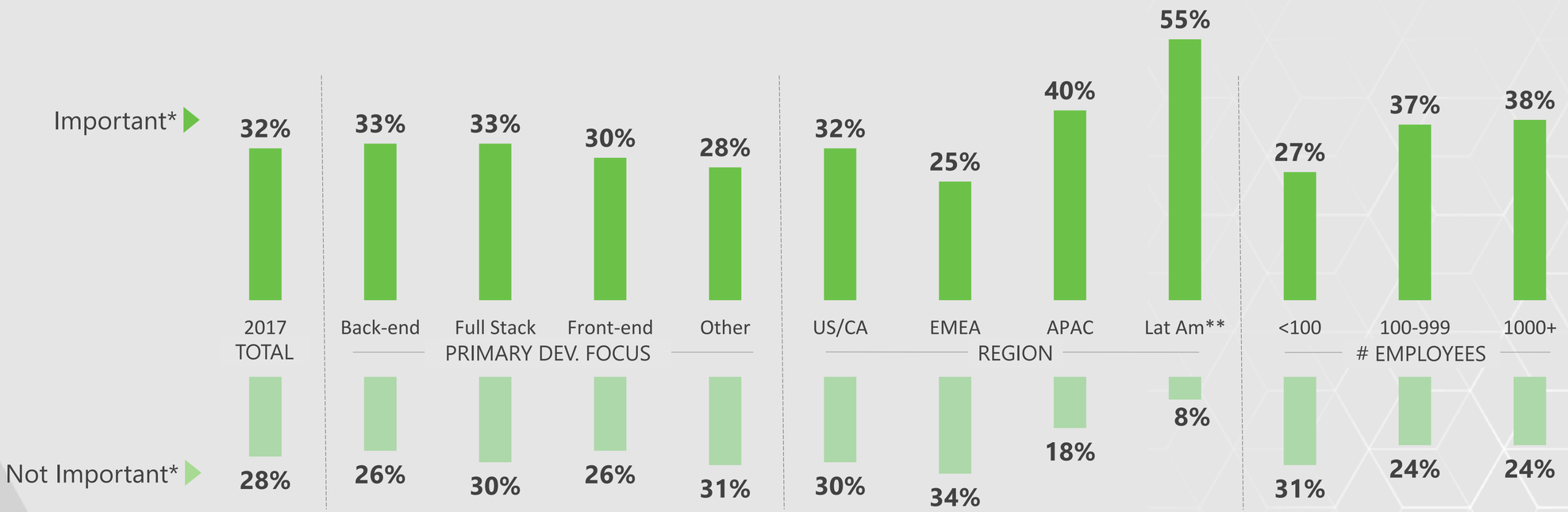


SOURCE: Q49, among those with an opinion; blanks excluded. * Top 3/Bottom 3 box on scale from 1 'not at all important to 10 'extremely important'.

Availability of Multiple Registries

- The availability of multiple registries is not widely seen as important – at least not outside Latin America.
- EMEA respondents, and those in companies with fewer than 100 employees are least likely to value access to multiple registries.

IMPORTANCE OF AVAILABILITY OF MULTIPLE REGISTRIES *e.g., react vs react-native or cli*



SOURCE: Q50, among those with an opinion; blanks excluded. * Top 3/Bottom 3 box on scale from 1 'not at all important to 10 'extremely important'. **Sample size small (n<50)

SECTION HIGHLIGHTS

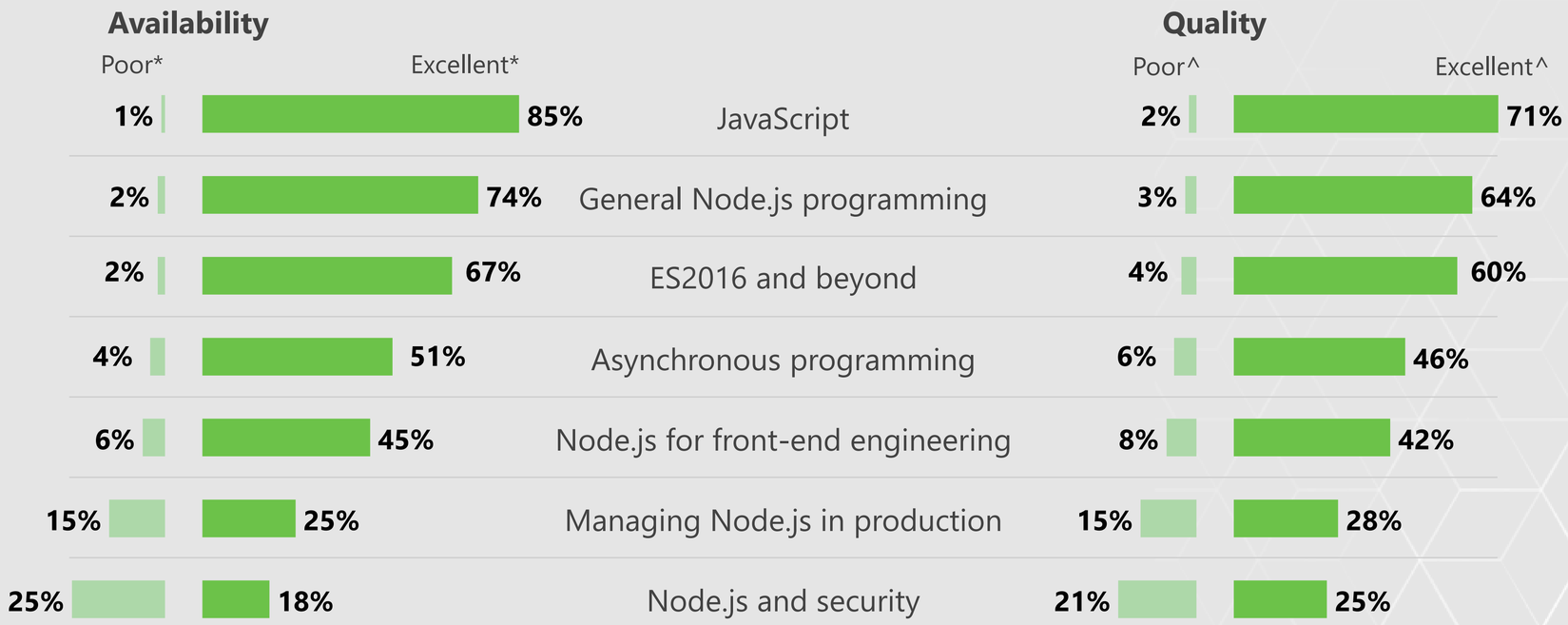
Learning Node.js

- The main way Node.js users learn a new language is through online courses without an instructor, especially outside the US/CA
- Nearly all users learned Node.js in English – but for more than half, it was not their native language.
- EMEA and Latin American users are most likely to have learned in a non-native language.
- There has been a rise in those who say it is easy to learn Node.js, and improvement in scores for availability and quality of resources in several topic areas.
- Latin American users, despite having learned in non-native language, give particularly high scores for availability, quality and overall ease of learning.
- There are some differences perceptions by subgroup, with APAC and EMEA having some concerns, and mid-size companies perhaps faring better than others.
- Newer Node.js users are less enthusiastic about availability and quality of resources than longer term users, although most are still positive.
- Documentation and StackOverflow are the main sources users rely on when learning a new language – but free online courses and tutorial videos are also important and something users would like more of (especially new users and those in Latin America.

Learning resources

- There have been notable improvements in access to and/or quality of learning resources – particularly for General Node.js programming and Asynchronous programming
- Still, more needs to be done to improve ratings for resources around managing Node.js in production and Node.js and security – for both areas, high negative scores are a red flag

AVAILABILITY / QUALITY OF LEARNING RESOURCES



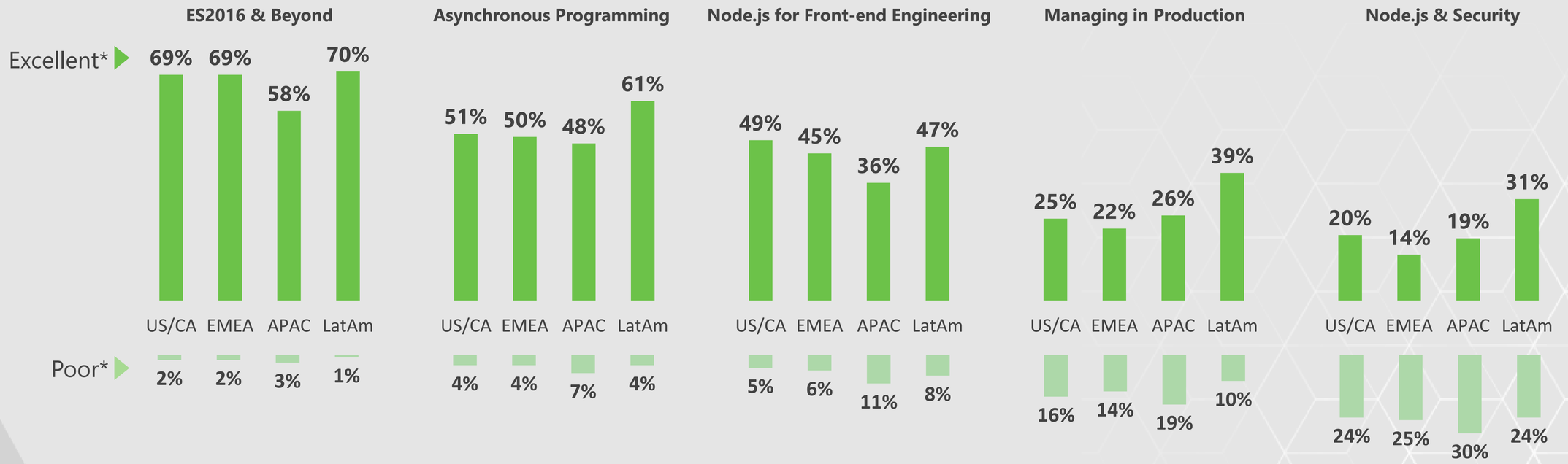
SOURCE: Q35, Q36, among those who provided a rating.

***Top 3/Bottom 3 on scale from 1 'poor – e.g., impossible to find/access' to 10 'excellent – e.g. easy to find/access'**
^ Top 3/Bottom 3 on scale from 1 'poor' to 10 'excellent'

Learning resources

- Perceptions of availability of resources vary considerably by region – with those in Latin America generally more upbeat, but those in APAC and EMEA having concerns in some specific topic areas

AVAILABILITY OF LEARNING RESOURCES BY REGION *selected resources*

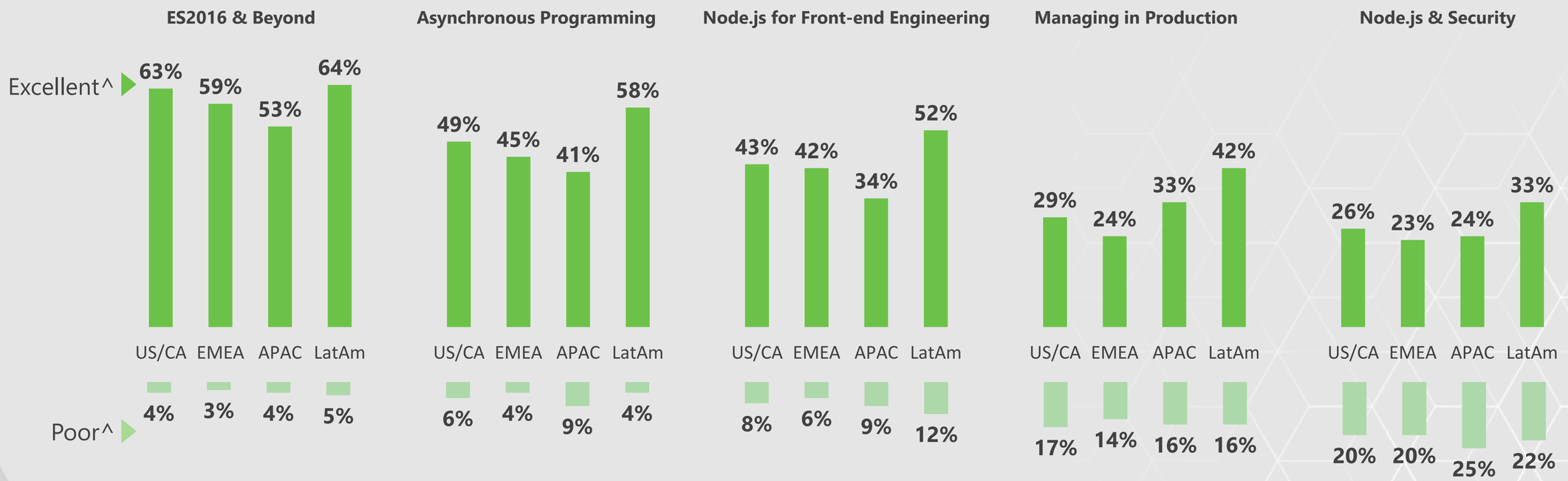


SOURCE: Q35 among those who provided a rating. ***Top 3/Bottom 3 on scale from 1 'poor – e.g., impossible to find/access' to 10 'excellent – e.g. easy to find/access'**

Learning resources

- A similar pattern exists with regard to quality: Latin America respondents are more pleased with quality while APAC and (less so) EMEA have concerns

AVAILABILITY OF LEARNING RESOURCES BY REGION *selected resources*

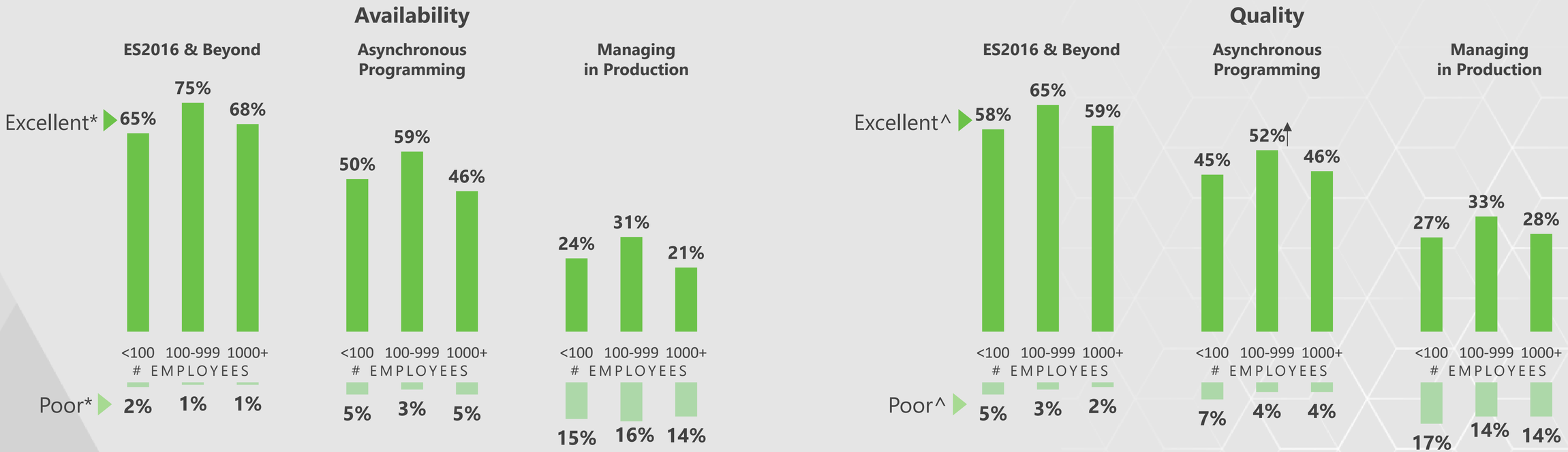


SOURCE: Q36 among those who provided a rating. **^Top 3/Bottom 3 on scale from 1 'poor' to 10 'excellent'**

Learning resources

- Respondents in mid-size companies perceive greater access to some learning resources than those in larger and smaller firms, but perceived quality is only marginally better
- To the extent there has been an improvement in perceptions of availability of resources, it's coming from midsize and smaller companies

AVAILABILITY / QUALITY OF LEARNING RESOURCES BY CO SIZE *selected resources*



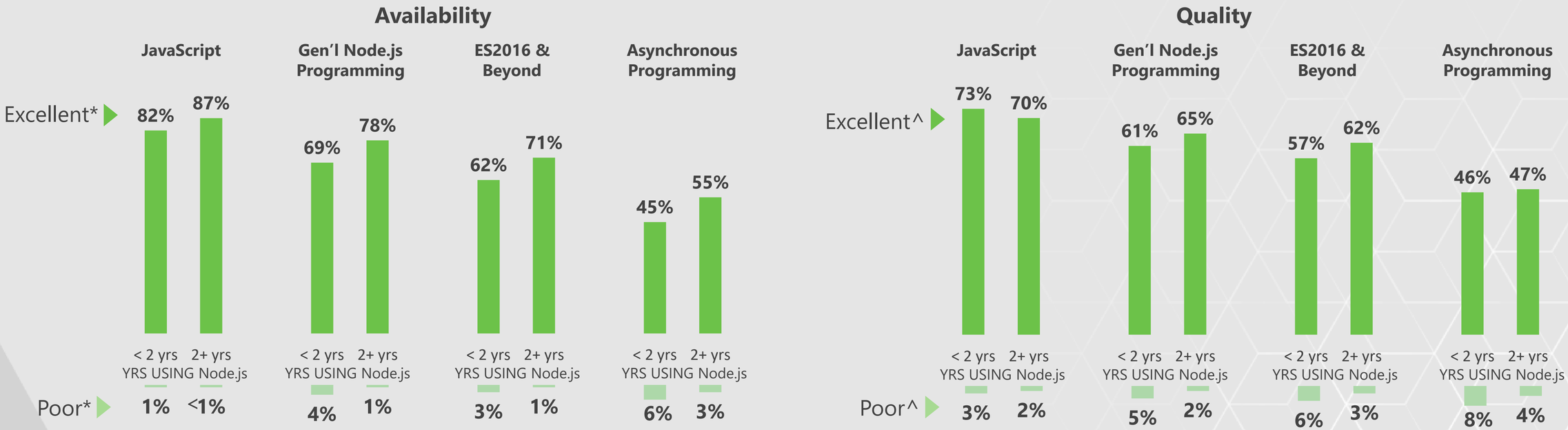
SOURCE: Q35, Q36, among those who provided a rating.

*Top 3/Bottom 3 on scale from 1 'poor – e.g., impossible to find/access' to 10 'excellent – e.g. easy to find/access'
^ Top 3/Bottom 3 on scale from 1 'poor' to 10 'excellent'

Learning Resources

- Newer Node.js users give lower scores than others on availability in several topic areas
- While still low, their higher negative ratings are worthy of note

AVAILABILITY / QUALITY OF LEARNING RESOURCES BY USING NODE.JS *selected resources*



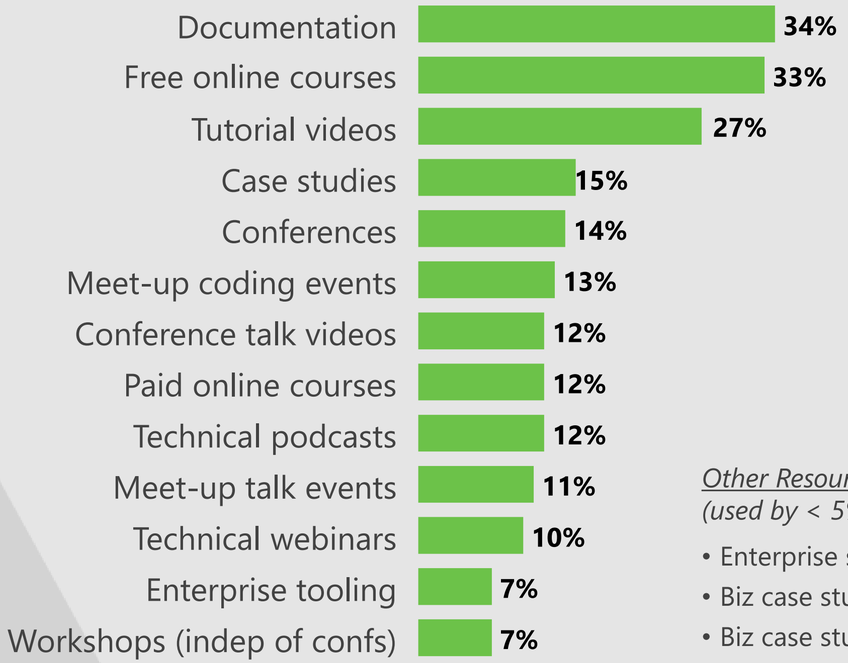
SOURCE: Q35, Q36, among those who provided a rating.
NOT TRENDED BY SUBGROUP

***Top 3/Bottom 3 on scale from 1 'poor – e.g., impossible to find/access' to 10 'excellent – e.g. easy to find/access'**
^ Top 3/Bottom 3 on scale from 1 'poor' to 10 'excellent'

Resources used

- To the extent users want more learning resources, documentation, free online courses and tutorial videos top the list
- Consistent with their current usage, new Node.js users are particularly likely to want free online courses and tutorial videos

RESOURCES WOULD LIKE MORE OF *top mentions*



Other Resources
(used by < 5%)

- Enterprise services
- Biz case study podcasts
- Biz case study webinars

RESOURCES WOULD LIKE MORE OF BY USING NODE.JS* *top mentions*

	< 2 yrs	2+ yrs
Documentation	**33%**	**34%**
Free online courses	**42**	**27**
Tutorial videos	**30**	**25**
Case studies	**15**	**16**
Conferences	**12**	**16**
Meet-up coding events	**14**	**12**
Paid online courses	**14**	**10**
Technical podcasts	**12**	**12**

SOURCE: Q38, including no answers

***Not trended by subgroup**

Resources used

- Latin American users are particularly open to new learning resources – particularly around free online courses, tutorial videos, conferences and (increasingly) conference talk videos
- Front end developers are less likely than others (and than last wave) to want more meet-up events

RESOURCES WOULD LIKE MORE OF *top mentions*

By Region					By Primary Development Focus			
US/CA	EMEA	APAC	LatAm		Back-End	Full Stack	Front-End	Other
34%	**34%**	**34%**	**36%**	Documentation	**40%**	**38%**	**35%**	**34%**
32	**31**	**36**	**46**	Free online courses	**39**	**38**	**38**	**35**
27	**24**	**30**	**39**	Tutorial videos	**31**	**35**	**27**	**23**
14	**12**	**15**	**29**	Conferences	**18**	**18**	**12**	**10**
11	**16**	**22**	**21**	Case studies	**19**	**17**	**16**	**15**
13	**10**	**17**	**17**	Meet-up coding events	**16**	**15**	**9**	**13**
11	**9**	**14**	**17**	Meet-up talk events	**13**	**14**	**8**	**10**
15	**10**	**12**	**17**	Technical podcasts	**18**	**14**	**10**	**7**
12	**10**	**11**	**22**	Conference talk videos	**14**	**13**	**11**	**12**

SOURCE: Q38, including no answers

SECTION HIGHLIGHTS

Node.js Versions & LTS

- Most Node.js users use a version manager – typically Nvm.
- Just over half use LTS release line, but use of current release line is increasing – particularly among Full stack and “other” developers, and in smaller companies.
- It is important to most users to have LTS for Node.js, although it is somewhat less important to those in small companies or in EMEA.
- There has been a drop in those who say the LTS schedule support timeframe is ‘clear’, down to just half of users. Those least likely to see as clear are front-end developers, APAC and those in smaller companies.
- New users are less likely to use a version manager, and are more likely to use Apt-Get than their more seasoned peers.
- Importantly, only a minority of new users report a good understanding of the LTS schedule/support timeframe.

More than half of users rely on LTS release line – but that number is slipping

Current is particularly popular among small companies, and newer Node.js users

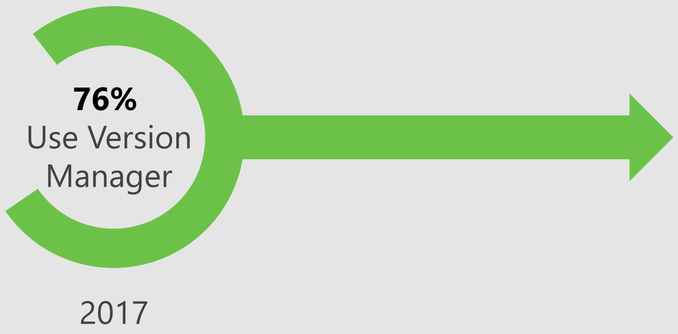
Release Line

SOURCE: Q39

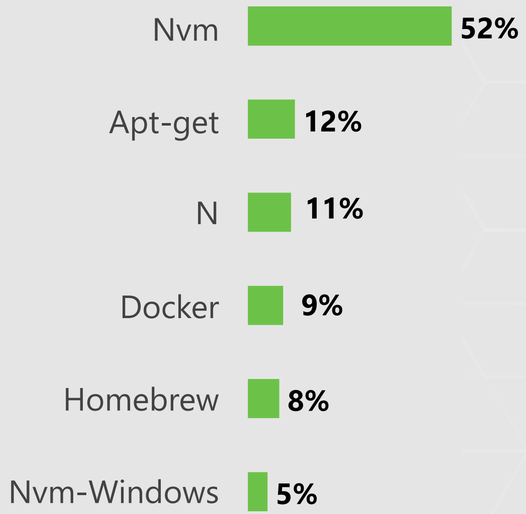
Version managers

- Three in four Node.js users say they use a Node.js version manager – typically NVM

USE A NODE.JS VERSION MANAGER



NODE.JS VERSION MANAGER USED *top mentions, among those who use any*



Other Version Managers (used by 1% or less).

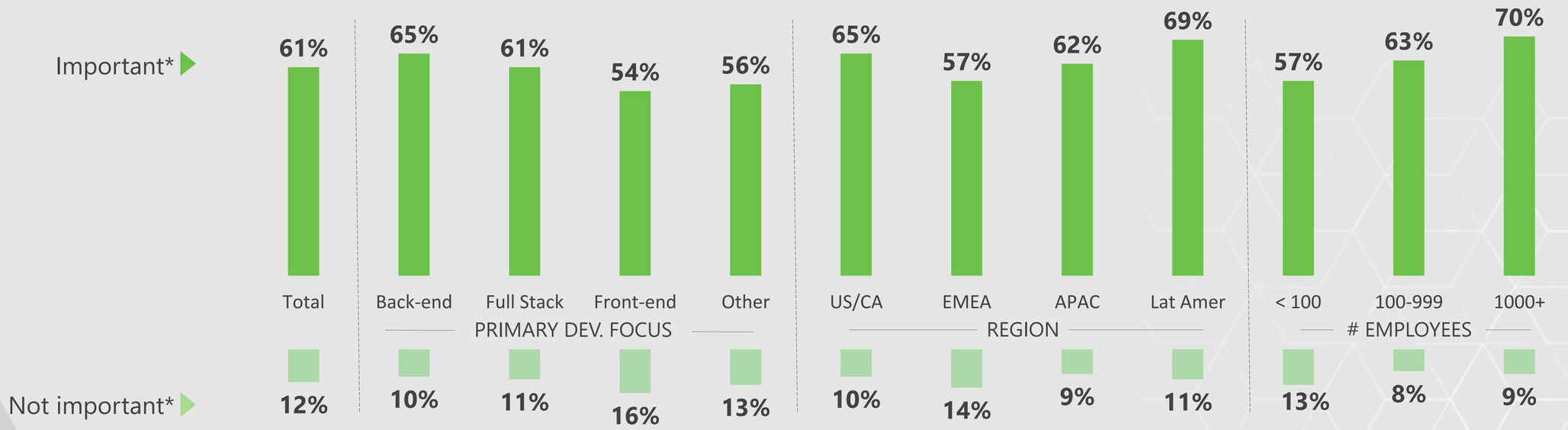
- Nodenv
- Nave
- Nodebrew
- Nodist

SOURCE: Q43

LTS support

- It is important to most users to have LTS for Node.js
- Users in EMEA and smaller companies are less likely to see it as a priority, but even here, more than half say it is important

IMPORTANCE OF HAVING LONG TERM SUPPORT (LTS) FOR NODE.JS

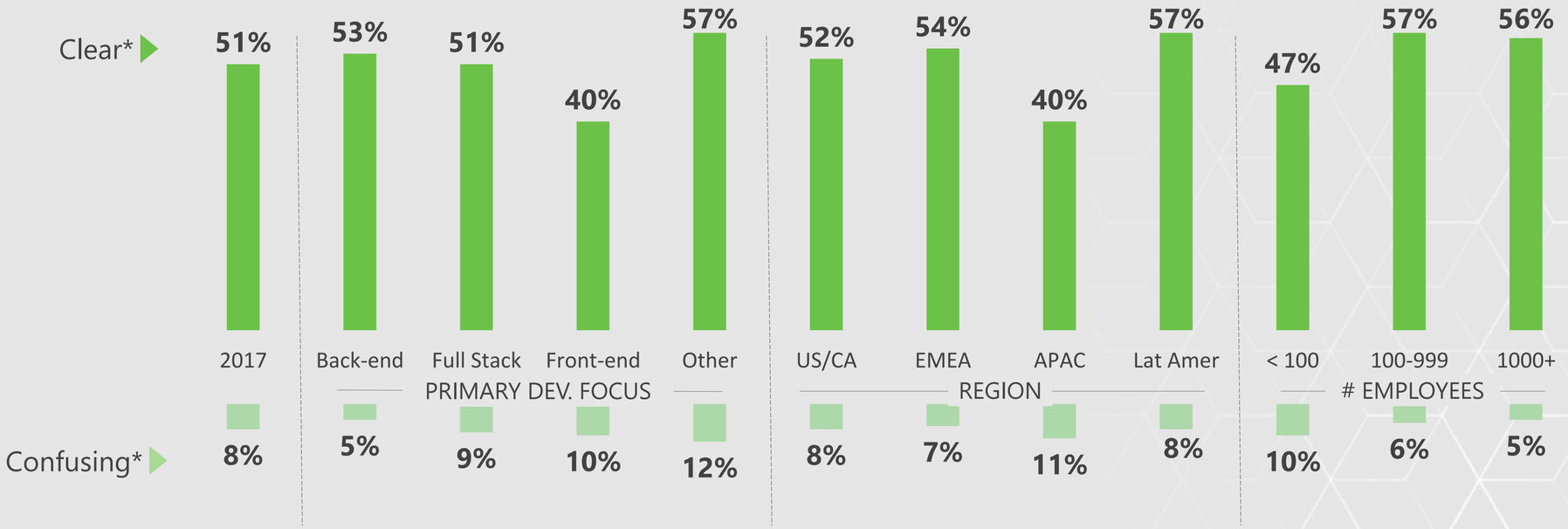


SOURCE: Q40, among those with an opinion; blanks excluded. * Top 3/Bottom 3 box on scale from 1 'not at all important' to 10 'extremely important' Question new in 2017/not trended

LTS support

- While many say the LTS Schedule / Support timeframe is clear, that number has dropped significantly since last year – overall and across multiple segments
- APAC users, front-end developers and those in small companies are least likely to see LTS schedule / support timeframe as clear

IMPORTANCE OF LTS SCHEDULE / SUPPORT TIMEFRAME FOR VARIOUS VERSIONS *by subgroup*



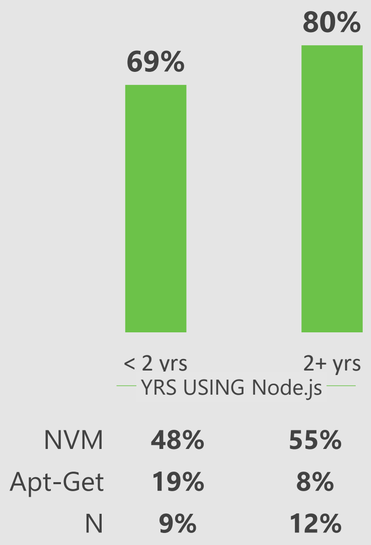
SOURCE: Q45, Among those with an opinion (Blanks excluded)

*Top 3/Bottom 3 box on scale from 1 'very confused' to 10 'perfectly clear'

Understanding newer Node.js users

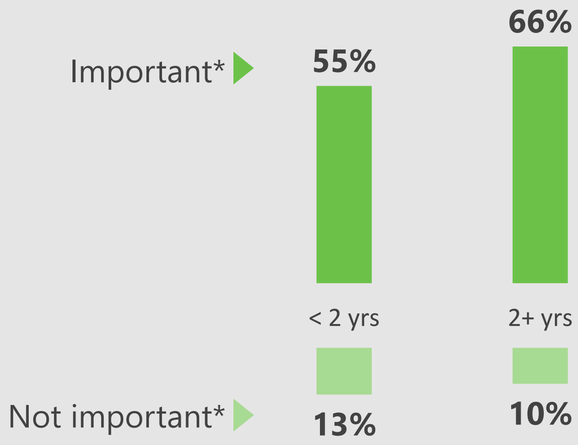
- Newer Node.js users are less likely than their more seasoned peers in their use of version manager, and are more likely to use Apt-Get
- While it is not as important to them to have LTS support, most novice users still do want it, and their understanding of the schedule/support timeframe is weak

USE VERSION MANAGER

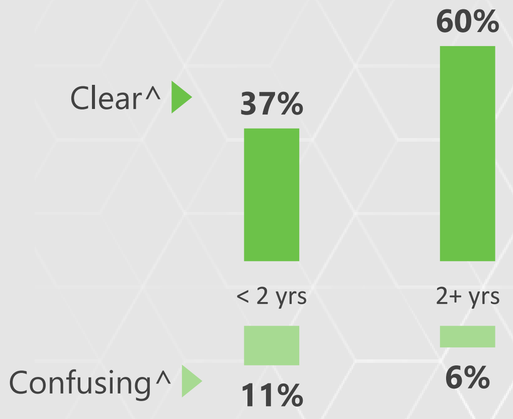


	< 2 yrs	2+ yrs
NVM	48%	55%
Apt-Get	19%	8%
N	9%	12%

IMPORTANCE OF HAVING LTS FOR NODE.JS



UNDERSTANDING OF LTS SCHEDULE / SUPPORT TIMEFRAME



SOURCE: Q40, Q43, Q45
NOT TRENDED BY SUBGROUP

*** Top 3/Bottom 3 box on scale from 1 'not at all important' to 10 'extremely important'**
^ Top 3/Bottom 3 box on scale from 1 'very confused' to 10 'perfectly clear'

SECTION HIGHLIGHTS

Node.js Impact & Getting Involved

- Users are very upbeat and excited about Node.js – with words like “fast” “easy” “awesome” “simple” “powerful” and “fun” widely used to describe Node.js
- Node.js is continuing to have a positive impact on many users – primarily through increased productivity and satisfaction, reduced development costs and increased app performance.
- The impacts may not be immediately clear however: new users are less likely to report positive impacts in many areas.
- While it’s not the most widely felt benefit, users in US/CA are more likely than others to say Node.js.js has helped with recruiting.
- Despite their positive perceptions, few have been contributing to open source projects for Node.js.
- There is growing interest in getting involved, however: nearly a third say they are interested in contributing and nearly half say they might be open to mentoring others (both up from last year).
- Those most interested in being involved include users in Latin America, APAC, and back-end and full stack developers.
- The main barriers are time and inexperience – but some Node.js users don’t know how to contribute, or feel the community is not welcoming.

Words to describe Node.js

In their own words, respondents used mostly positive adjectives to describe Node.js

They particularly like that it is fast, simple, easy, yet powerful and flexible.

SOURCE: Q53

Business impact

- Users in Latin America are particularly likely to note positive impacts – particularly around productivity, application performance and uptime
- Node.js has helped with recruiting in the US/CA more than other regions

HOW NODE.JS HAS IMPACTED YOUR BUSINESS *top impacts*

	US/CA	EMEA	APAC	LAT AM
Increased developer productivity	**68%**	**66%**	**69%**	**78%**
Improved developer satisfaction	**64**	**60**	**60**	**65**
Reduced development costs	**60**	**51**	**61**	**56**
Increased application performance	**43**	**47**	**53**	**63**
Increased uptime	**22**	**23**	**24**	**40**
Helped recruit developers	**34**	**18**	**18**	**12**
No impact	**7**	**6**	**3**	**3**

SOURCE: Q54

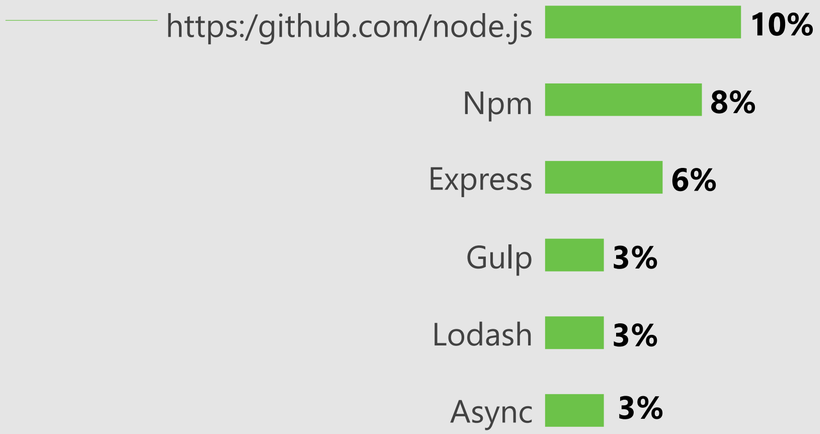
Groups working with

- Consistent with last year, relatively few Node.js users reports contributing to open source repositories
- Github, npm and Express are most widely contributed to

OPEN SOURCE REPOSITORIES / GROUPS / PROJECTS IN NODE.JS CONTRIBUTE TO

Most Widely Mentioned:

- ...nodejs/nodejs.org **3%**
- ...nodejs/LTS **2%**
- ...nodejs/v8 **2%**
- ...nodejs/http2 **1%**
- ...nodejs/build **1%**
- ...nodejs/node-gyp **1%**
- ...nodejs/TSC **1%**



2% or fewer mention. . .

- Request
- NodeSchool
- Browserify
- I18n
- Libuv
- Streams
- Nodebots
- Node-serialport

SOURCE: Q19

Groups working with

- Users in Asia / Pacific region are more likely to contribute to many groups

OPEN SOURCE REPOSITORIES / GROUPS / PROJECTS IN NODE.JS CONTRIBUTE TO

By Region

US/CA	EMEA	APAC	LatAm	
12%	**7%**	**12%**	**9%**	https:/github.com/node.js
5	**7**	**14**	**9**	Npm
4	**5**	**11**	**6**	Express
2	**3**	**5**	**4**	Gulp
2	**3**	**6**	**4**	Lodash
1	**3**	**6**	**4**	Async
1	**1**	**4**	**3**	Browserify

By Primary Development Focus

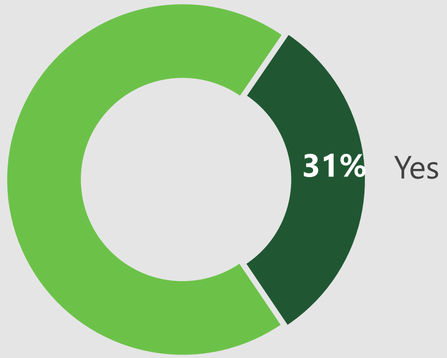
Back-End	Full Stack	Front-End	Other
10%	**10%**	**5%**	**12%**
6	**7**	**8**	**12**
6	**7**	**4**	**7**
3	**3**	**4**	**4**
2	**3**	**3**	**3**
3	**3**	**1**	**4**
2	**1**	**2**	**2**

SOURCE: Q19

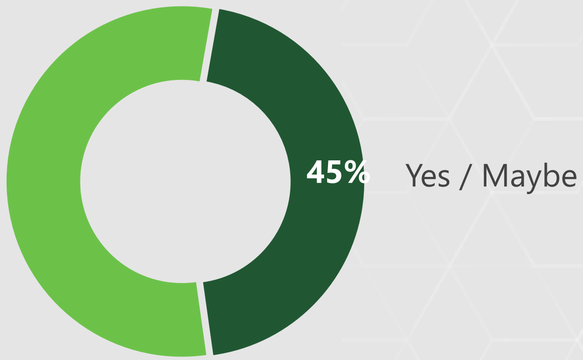
Interest in getting involved

- Encouragingly, there has been a rise in the number of Node.js users who are interested in contributing to the project and/or mentoring others
- Those most interested in being involved include users in Latin America and APAC, and back-end and full stack developers
- While time and lack of skill are, by far, the primary barriers to contributing, some are held back because they don't know HOW to contribute

INTEREST IN CONTRIBUTING TO OPEN SOURCE NODE.JS PROJECT



INTEREST IN MENTORING OTHERS



SOURCE: Q58, Q59, Q60