CNCF TECHNOLOGY RADAR

OBSERVABILITY TECHNOLOGIES & API MANAGEMENT AND DEVELOPER EXPERIENCE

CLOUD NATIVE
COMPUTING FOUNDATION

MARCH 2025

Can I share data from this report?

1. License Grant

2. Limitation of Liability

SlashData^{TM}, believes the statements contained in this publication to be based upon information that we consider reliable, but we do not represent that it is accurate or complete and it should not be relied upon as such. Opinions expressed are current opinions as of the date appearing in this publication only and the information, including the opinions contained herein, are subject to change without notice. Use of this publication by any third party for whatever purpose should not and does not absolve such third party from using due diligence in verifying the publication's contents. SlashData^{TM} disclaims all implied warranties, including, without limitation, warranties of merchantability or fitness for a particular purpose.

SlashData^{TM}, its affiliates, and representatives shall have no liability for any direct, incidental, special, or consequential damages or lost profits, if any, suffered by any third party as a result of decisions made, or not made, or actions taken, or not taken, based on this publication.

The analyst of the developer economy | formerly known as VisionMobile

ABOUT THE AUTHORS

Liam Bollmann - Dodd

Senior Market Research Consultant

Liam is a former experimental antimatter physicist, and he obtained a PhD in Physics while working at CERN. He is interested in the changing landscape of cloud development, cybersecurity, and the relationship between technological developments and their impact on society.

✉ liam.dodd@slashdata.co

Nikita Solodkov

Market Research and Statistics Consultant

Nikita Solodkov is a multidisciplinary researcher with a particular interest in using data-driven insights to solve real-world problems. He holds a PhD in Physics and has over five years of experience in data analytics and research design.

✉ nikita.solodkov@slashdata.co

TABLE OF CONTENTS

KEY INSIGHTS

- For API Management and Developer Experience tools, Backstage and DevSpace were considered technologies ready to 'adopt'. →

- Backstage was considered the most mature, most useful, and the technologies that developers were most likely to recommend. →

- Carvel is currently an early-stage project, but achieved high scores for its usefulness, and was highly recommended by other developers. →

- For Observability technologies, Cortex and OpenTelemetry were placed in the 'adopt' position on the technology landscape radar. →

- Despite OpenTelemetry receiving leading scores for its maturity and usefulness, fewer developers indicated they would recommend it than many other technologies. →

- Security concerns ranked as the most common challenge across both Observability technologies and API Management and Developer Experience tools. However, a third of developers using Observability tools stated they had no major challenges. →

- Developers who do not experiment with cloud native technologies, at early stages, were much more likely to highlight challenges with a lack of institutional support than those who experiment. →

- The perception of many technologies differed substantially between those who experimented with cloud native technologies compared to those who didn't, highlighting the challenges of meeting starkly meeting different audiences where they are. →

INTRODUCTION

01

1. Introduction

In Q1 2025, more than 320 professional developers using technologies associated with cloud-native development were asked about their experience and opinions with regard to observability technologies and platform tools for developer experience and API management. The technologies shown to developers were selected by CNCF and CNCF's End User Community for relevance and importance. The developers surveyed originate from around the world and have a large range of specialities and areas of focus. A more granular breakdown of the respondents is included in the Methodology section.

For the products or tools they were familiar with, they rated them on their usefulness and maturity and indicated how likely they were to recommend that technology to other developers. Within the context of this report, usefulness was defined as how well a given technology meets project requirements, and maturity was related to its stability and reliability. The recommendation scale was converted into a net promoter score (NPS) for use during the analysis.

Based on the usage, usefulness and maturity ratings, and how likely they are to recommend a given technology, we categorised the technologies into four groups: adopt, trial, assess, and hold. 'Adopt' technologies are considered reliable choices for most use cases, while 'trial' technologies are worth exploring to see if they meet your specific needs. 'Assess' technologies require careful evaluation before committing, and 'hold' technologies are considered less mature or useful in their current state.

Note: These radar position groups do not necessarily correlate with the CNCF maturity model (Sandbox, Incubating, and Graduated), which corresponds to the *Innovators, Early Adopters*, and *Early Majority* tiers from Geoffrey A. Moore's Crossing the Chasm: *Marketing and Selling High-Tech Products to Mainstream Customers.*

Sandbox: Sandbox projects are in their earliest stages, meant for experimentation and foundational growth. They are newer technologies that represent initial concepts and technologies with significant room for evolution.

Incubating: Projects that have a solidified technical vision and growing contributor base, but are still maturing in terms of community adoption, stability, and governance.

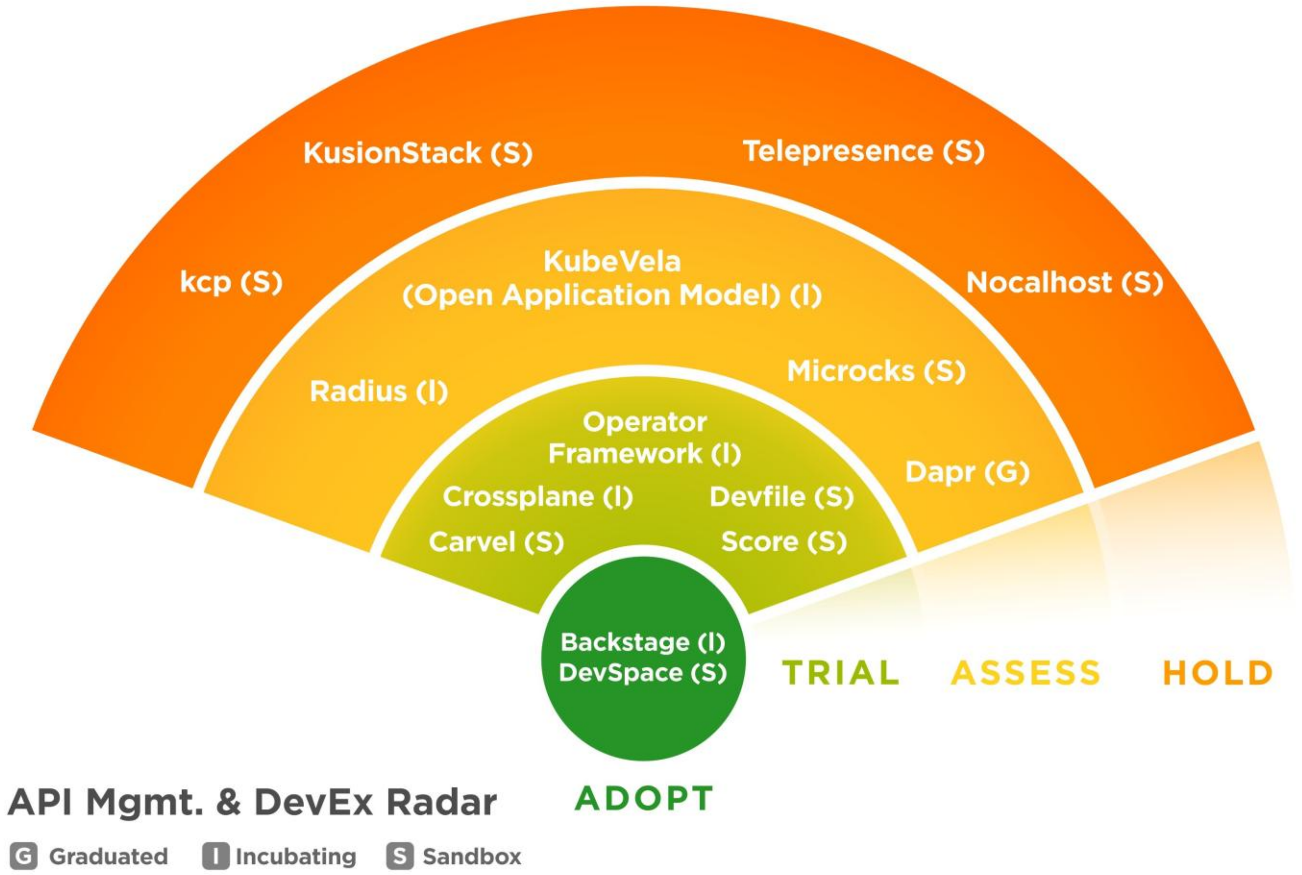
Graduated: Graduated projects are widely adopted and reliable. They have established a diverse community base supported by mature technical policies and governance.

API MANAGEMENT AND DEVELOPER EXPERIENCE TECHNOLOGIES

02

2. API Management and Developer Experience Technologies

For API Management (API Mgmt) and Developer Experience (DevEx) technologies, we see Backstage and DevSpace emerge as the two technologies that respondents cumulatively would place in the ‘adopt’ position of the technology radar. Dapr, the only Graduated project included in this survey wave is placed in the ‘assess’ position.



Developers familiar with API Mgmt & DevEx Tools (n=280)

Based on developer perceptions: 'adopt' technologies are considered reliable choices for most use cases, 'trial' technologies are worth exploring to see if they meet your specific needs, 'assess' technologies require careful evaluation before committing, and 'hold' technologies are considered less mature or useful in their current state.

** Graduated, Incubating, and Sandbox refer to [CNCF's hosted project](#) levels*

Usefulness

Backstage received the highest usefulness score (CSAT[1] of 92), the highest proportion of 5-star ratings (61%), and no 1-star or 2-star ratings. Carvel ranked second on usefulness score (84) but with a lower proportion of 5-star ratings than several technologies that received lower overall usefulness scores, such as DevSpace.

92%

OF DEVELOPERS FAMILIAR WITH BACKSTAGE RATE ITS MATURITY POSITIVELY

[1] *CSAT, or customer satisfaction, is the proportion of 4 and 5-star ratings, multiplied by 100, giving a scale of between 0 and 100.*

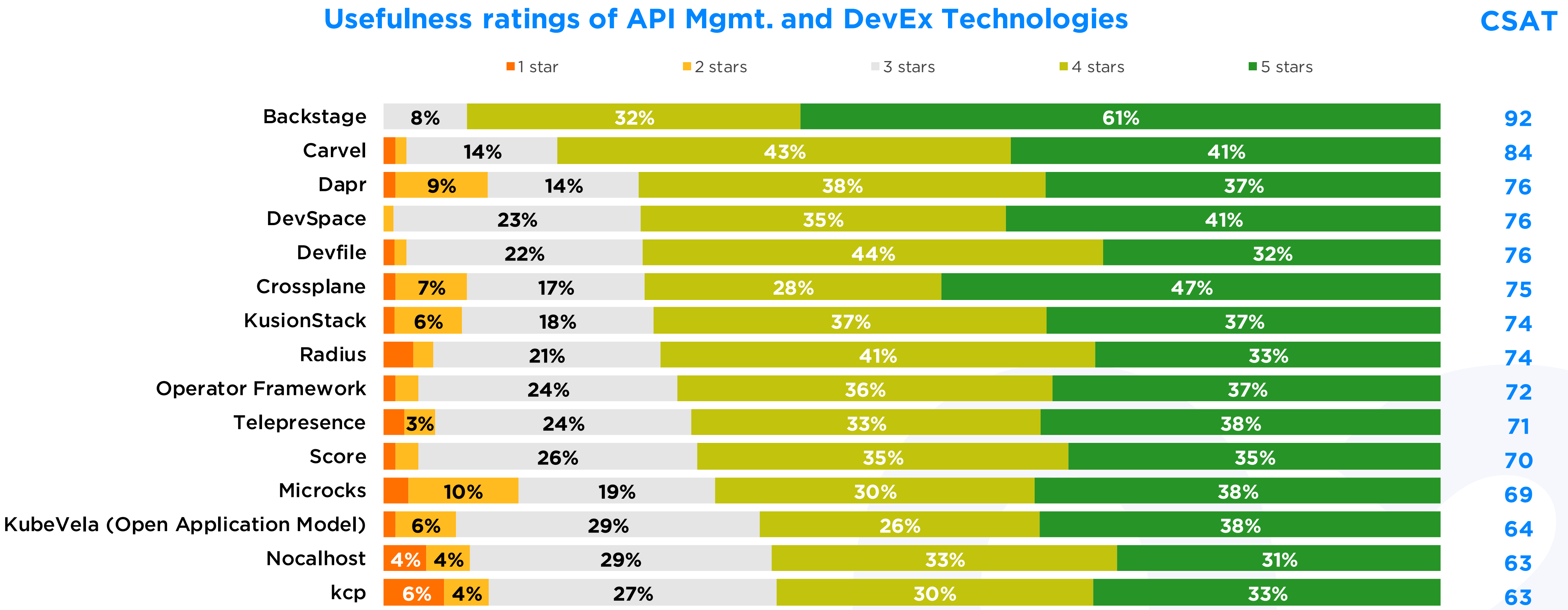
2. API Management and Developer Experience Technologies

Dapr received the third-highest usefulness score (76) but also received the second-highest proportion of 1-star and 2-star ratings (10%). Only Microcks got more negative ratings (12%) but also a much lower usefulness score (69). As such, Dapr appears to have high utility to most developers familiar with it, but for a small proportion it has developed a negative association.

Despite the difference in technology usefulness ratings, each technology received a majority of positive ratings, highlighting that developers typically consider the tools they are familiar with useful.

Usefulness ratings of API Management and Developer Experience Technologies

Usefulness ratings of API Mgmt. and DevEx Technologies



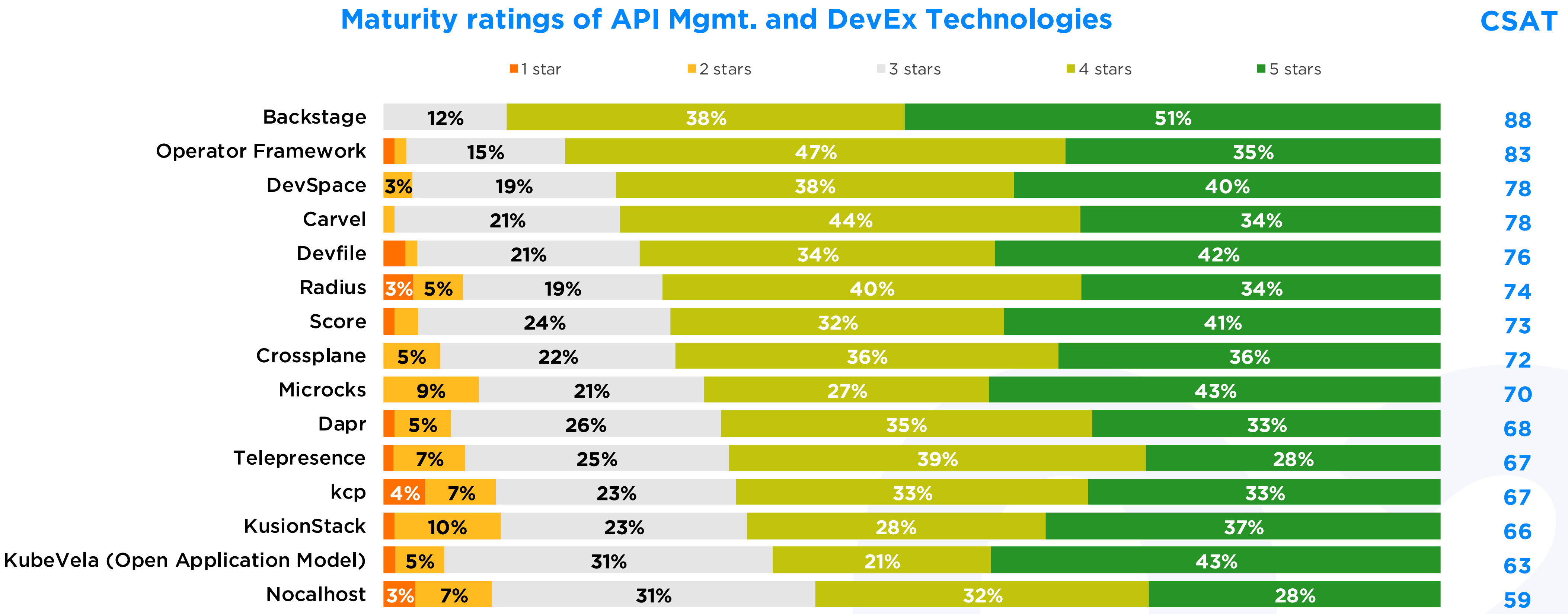
Question wording: How would you rate the following developer experience and API management tools with respect to these aspects? (Usefulness) % of developers familiar with the technology | CSAT (Customer satisfaction, is the proportion of 4 and 5-starratings, multiplied by 100, giving a scale of between 0 and 100) (n=280)

Maturity

For the maturity of the technologies, Backstage also received high ratings here, with a maturity score of 88, and 51% of developers familiar with Backstage rated its maturity as 5 stars. Operator Framework received the second-highest maturity score (83), but received fewer 5-star ratings (33%) than many technologies that received lower maturity scores overall. A stark example can be seen with the ratings that KubeVela received.

KubeVela received the second lowest maturity score (63), but the joint second highest proportion of 5-star ratings (43%), behind only Backstage and equal to Microcks. KubeVela's comparatively low score is a result of nearly a third of respondents familiar with it (31%) providing a 3-star rating. In other words, it doesn't stand out as mature or immature to these developers, and may instead be leaving a minimal impression.

Maturity ratings of API Management and Developer Experience Technologies



Question wording: How would you rate the following developer experience and API management tools with respect to these aspects? (Maturity) % of developers familiar with the technology | CSAT (Customer satisfaction, is the proportion of 4 and 5-starratings, multiplied by 100, giving a scale of between 0 and 100) (n=280)

Likelihood to recommend

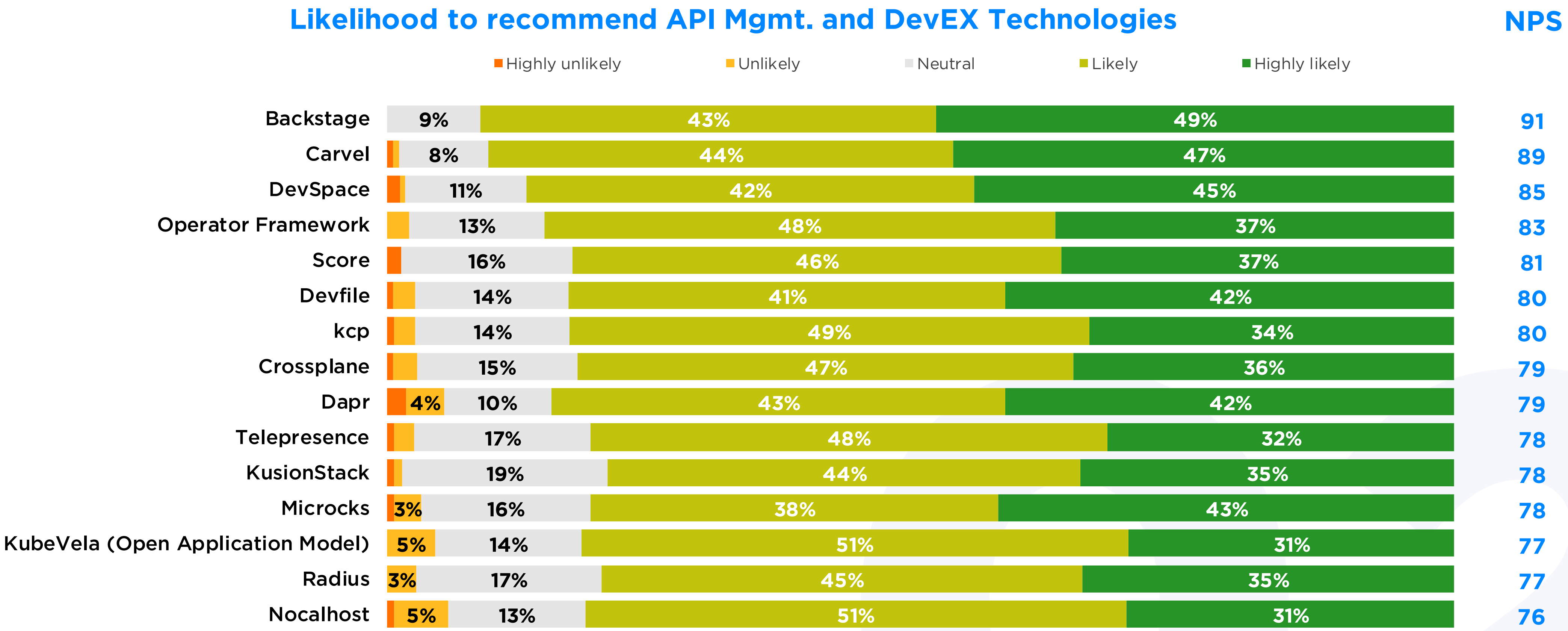
Backstage also takes the top spot for likelihood to recommend with a net promoter score (NPS) of 91, and 49% of respondents familiar with Backstage are highly likely to recommend it. Carvel takes the second spot, with an NPS of 89.

85% of respondents familiar with Dapr were either likely or highly likely to recommend it, but it also received the highest proportion of developers who were unlikely or highly unlikely to recommend it. Dapr received a similar split when respondents rated it for its maturity, suggesting that the concerns about its maturity may be leading a segment of the developer population to suggest against its use.

> ***For all technologies, the majority of respondents would recommend each of them***

Likelihood to recommend API Management and Developer Experience Technologies

Likelihood to recommend API Mgmt. and DevEX Technologies

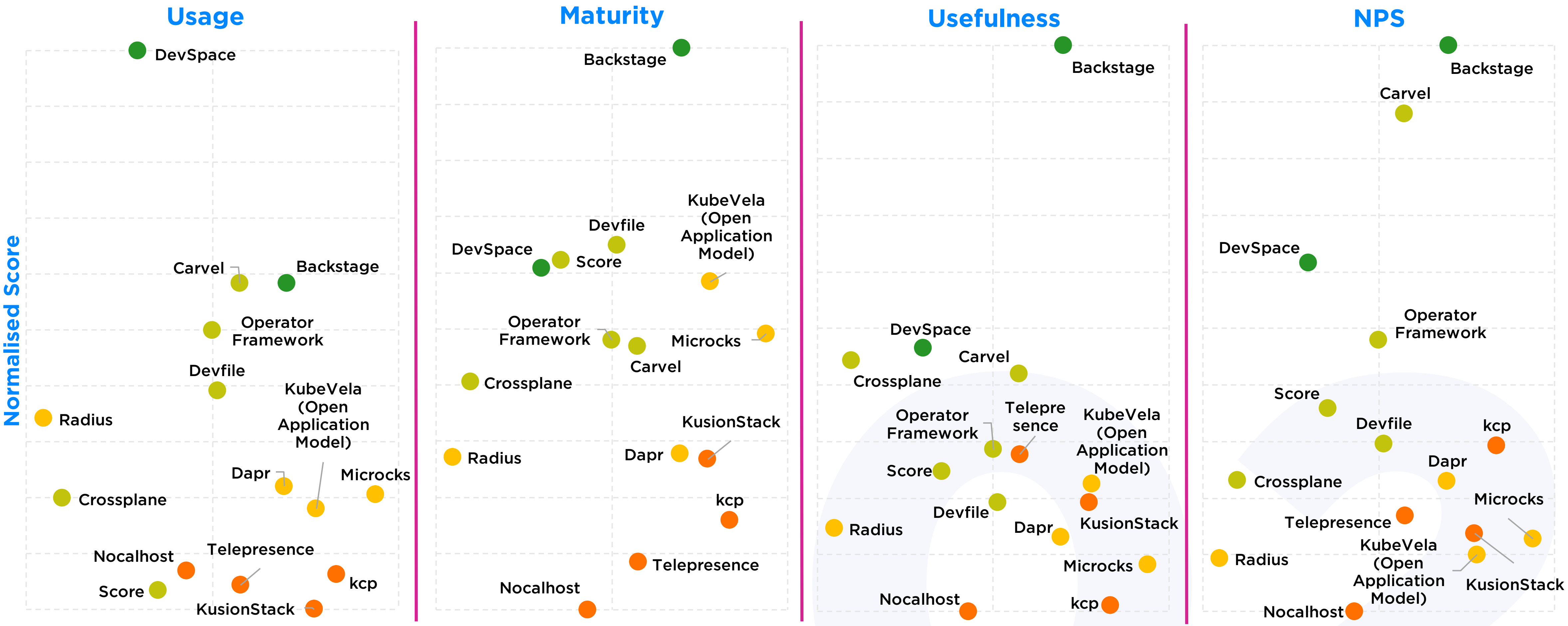


Question wording: How likely are you to recommend the following developer experience and API management tools?
% of developers familiar with the technology | NPS (Customer satisfaction, is the proportion of 4 and 5-starratings, multiplied by 100, giving a scale of between 0 and 100) (n=280)

Normalised scale of four variables

By using a normalised scale for each aspect of the technology considered for projecting their position on the technology radar, we can see the overall positioning of technologies by their radar position. In general, technologies in the adopt or trial position are more likely to receive higher normalised scores for each aspect, while those in the hold position receive lower scores.

Distribution of normalised score for API Mgmt. and DevEx technologies



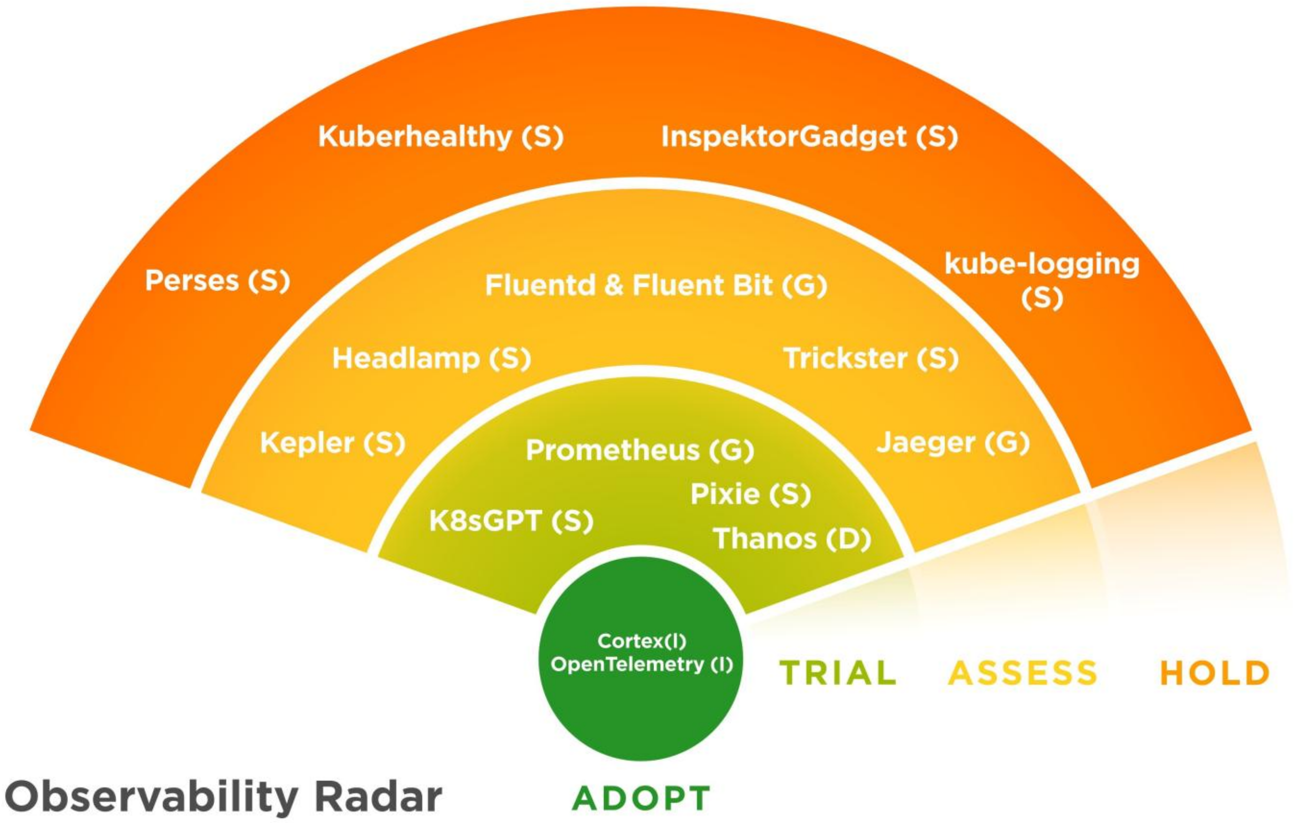
Legend: Dark Green: Adopt, **Light Green**: Trial, **Yellow:** Assess, **Orange:** Hold
Normalised score of usage, maturity, usefulness, and NPS (n=280)

OBSERVABILITY TECHNOLOGIES

03

3. Observability Technologies

For Observability technologies, Cortex and OpenTelemetry received high enough cumulative ratings across all of the factors to be considered for the 'adopt' position of the technology landscape radar. Prometheus was placed in the 'trial' position of the radar, the highest positioning of any of the Graduated technologies for this category.



Observability Radar

G Graduated   I Incubating   S Sandbox

Developers familiar with Observability technologies (n=273)

Based on developer perceptions: 'adopt' technologies are considered reliable choices for most use cases, 'trial' technologies are worth exploring to see if they meet your specific needs, 'assess' technologies require careful evaluation before committing, and 'hold' technologies are considered less mature or useful in their current state.

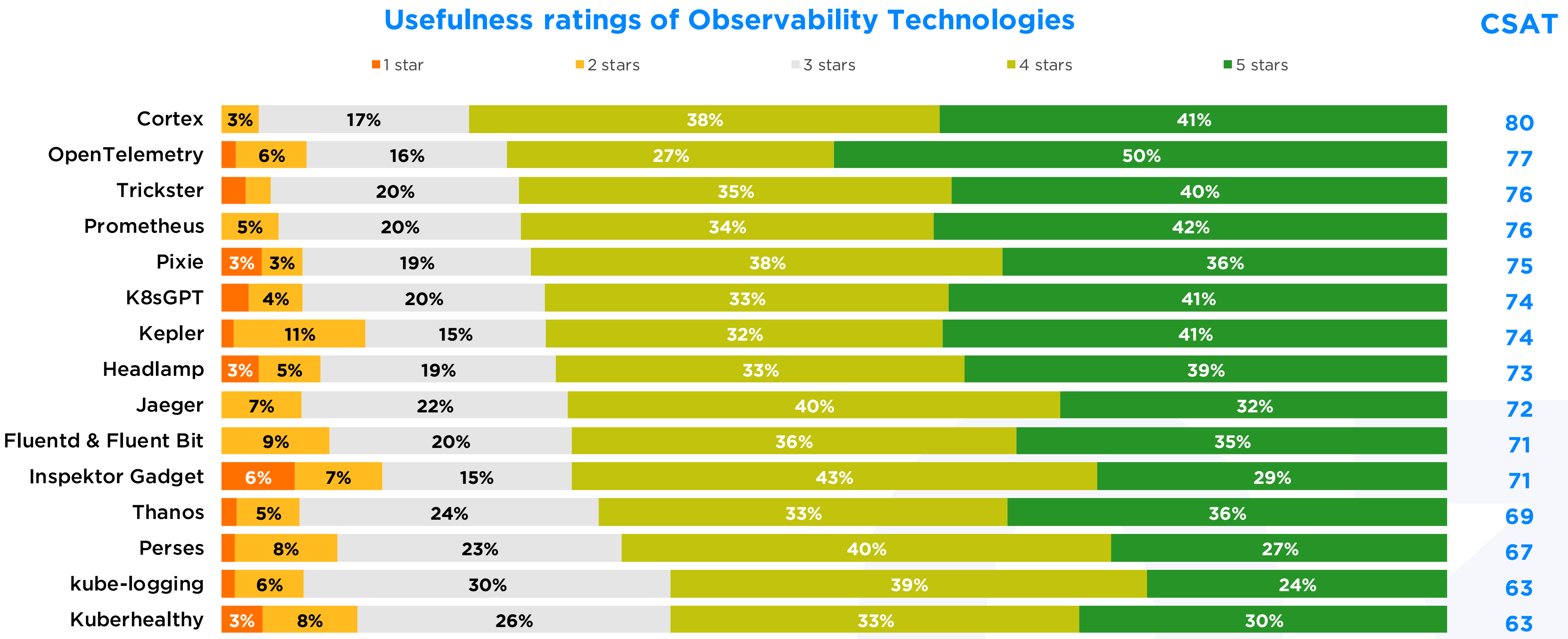
** Graduated, Incubating, and Sandbox refer to CNCF and LF AI & Data's hosted projects' levels*

Usefulness

Cortex received the highest usefulness score (CSAT of 80), and 41% of respondents familiar with Cortex gave it a 5-star rating for usefulness. OpenTelemetry received the second-highest usefulness score (77) but the greatest proportion of 5-star ratings (50%). However, OpenTelemetry also had a much higher proportion of 1-star and 2-star ratings (7%) than the other technologies with high usefulness scores.

Kepler and Inspektor Gadget received the highest proportions of 1-star and 2-star ratings, 12% and 13% respectively. Neither of these technologies have low usefulness scores, compared to the other technologies surveyed. This suggests that the negative views of their usefulness are related to the needs of a specific cohort of developers, who may find these technologies poorly suited to their needs.

Usefulness ratings of Observability Technologies



Usefulness ratings of Observability Technologies

Question wording: How would you rate the following observability tools with respect to these aspects? (Usefulness)
% of developers familiar with the technology | CSAT (Customer satisfaction, is the proportion of 4 and 5-starratings, multiplied by 100, giving a scale of between 0 and 100) (n=273)

Maturity

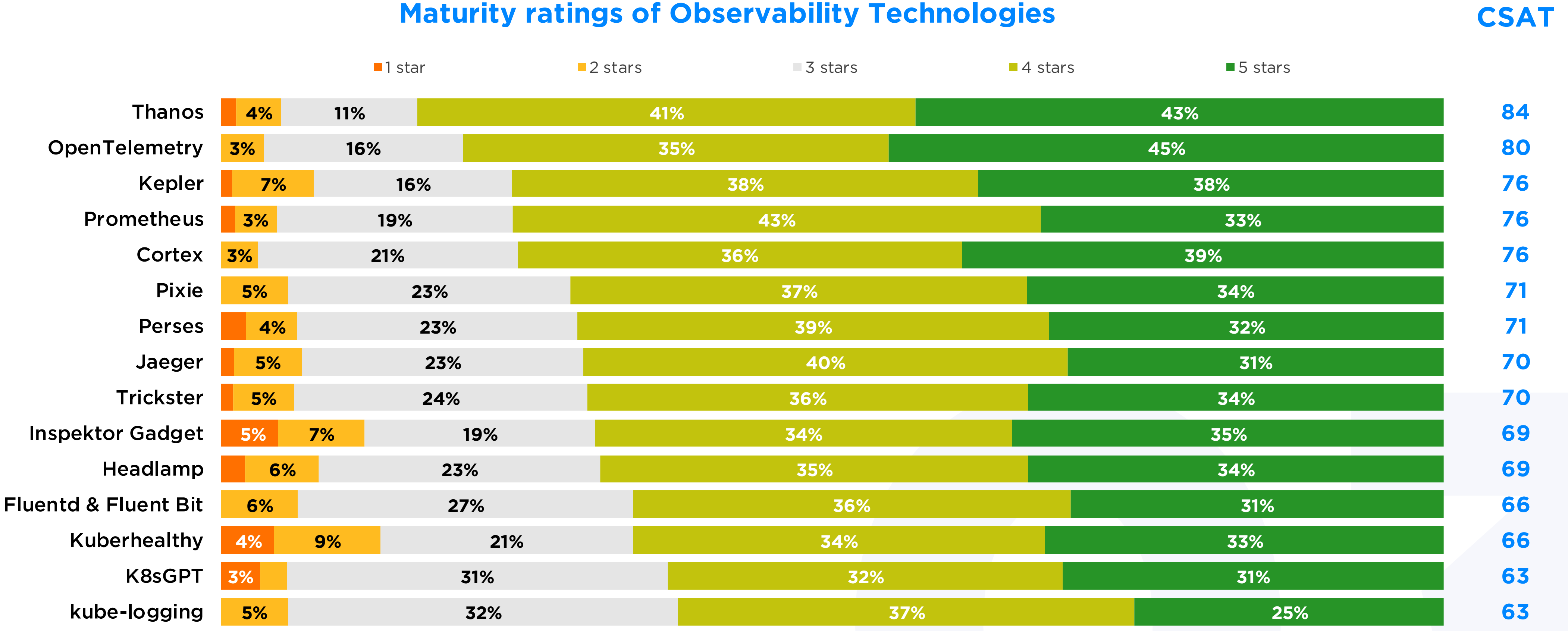
Thanos receives the highest score for maturity (CSAT of 84), with 42% of respondents familiar with it rating its maturity 5 stars. This high score is in contrast to Thano's usefulness score ranking, where it was the fourth lowest. This suggests that while respondents consider Thanos' features and capabilities to be well developed, it lacks the features that are of greatest utility to the developers surveyed.

OpenTelemetry had the second-highest maturity score (80), matching its ranking for usefulness. OpenTelemetry also received the highest proportion of 5-star ratings, 45%.

Inspektor Gadget and Kuberhealthy received the highest proportion of 1 and 2-star ratings for maturity, 12% and 13% respectively. Kuberhealthy scores low on maturity (66), and also was the lowest-rated technology on usefulness. This leaves Kuberhealthy as a technology that is both performing relatively poorly on usefulness and maturity, and disappointing the largest proportion of users on matters of technological maturity.

Maturity ratings of Observability Technologies

Maturity ratings of Observability Technologies



Question wording: How would you rate the following observability tools with respect to these aspects? (Maturity)
% of developers familiar with the technology | CSAT (Customer satisfaction, is the proportion of 4 and 5-starratings, multiplied by 100, giving a scale of between 0 and 100) (n=273)

Likelihood to recommend

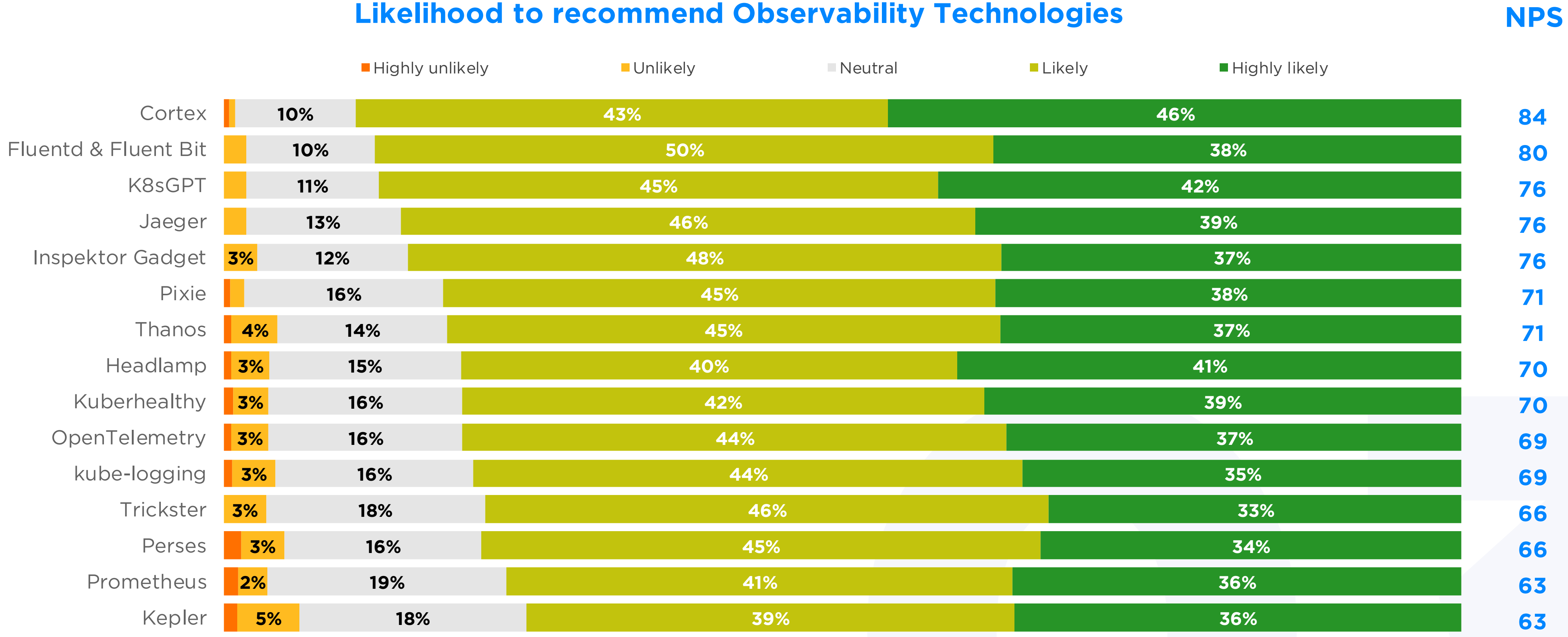
Cortex had the highest NPS (88), and the highest proportion of respondents who are highly likely to recommend it. Cortex also had the highest usefulness score and a high maturity score, which combined to lead to a technology those familiar with it are keen to recommend.

Fluentd & Fluent Bit had the second-highest NPS (86). However, this technology had a much smaller proportion of those who were highly likely to recommend it (38%) compared to other technologies with similarly high NPS. While Fluentd & Fluent Bit received scores for usefulness and maturity that were in the bottom half of both rankings, it still achieved an overwhelming rate of recommendation.

Despite Fluentd and Fluent Bit serving distinct roles in log processing, their shared ecosystem and overlapping use cases justify grouping them as a single technology. However, this aggregation may contribute to diverging results. As such, while criticisms or concerns of either of the tools may be associated with the lower scores in maturity and usefulness, together, they represent an ecosystem that developers are likely to recommend to others.

Likelihood to recommend Observability Technologies

Likelihood to recommend Observability Technologies

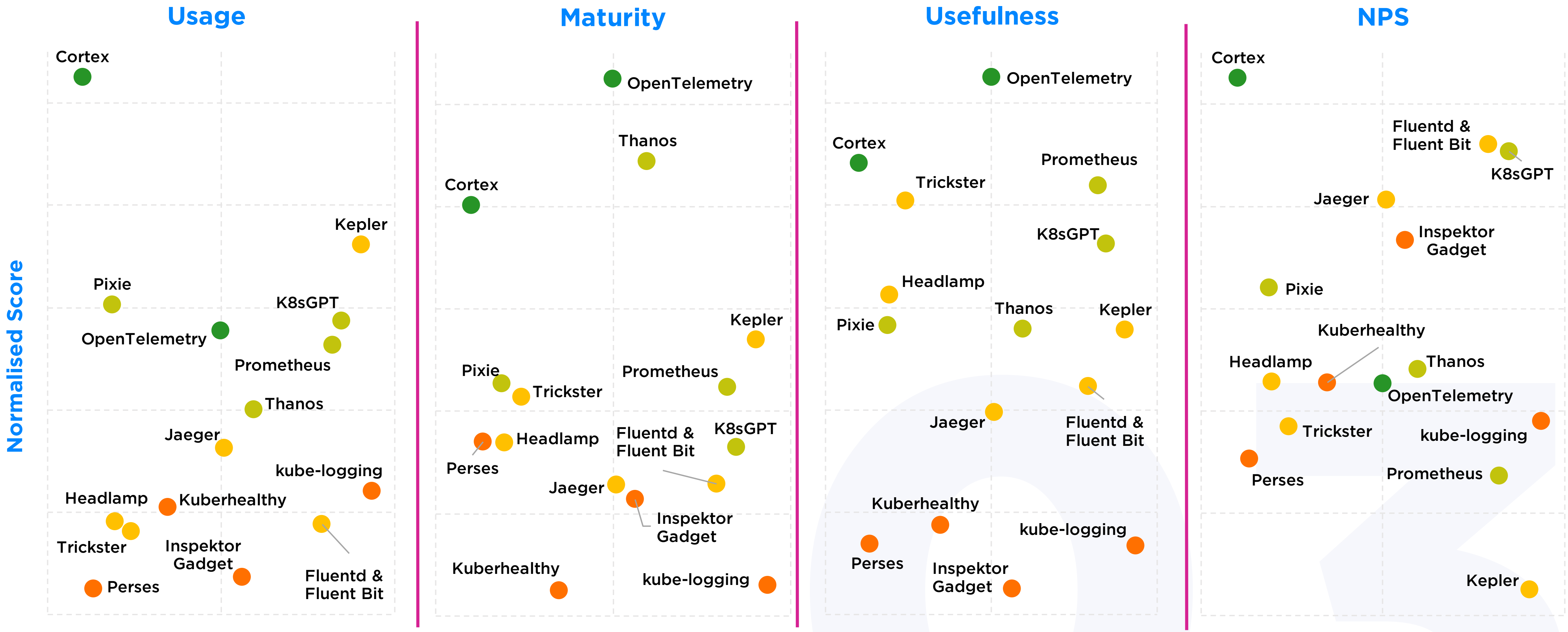


Question wording: How likely are you to recommend the following observability tools?
% of developers familiar with the technology | NPS (Customer satisfaction, is the proportion of 4 and 5-starratings, multiplied by 100, giving a scale of between 0 and 100) (n=273)

Normalised scores of four variables

Comparing the normalised scores across all four features that impact each technology's position on the technology landscape radar exposes some interesting behaviours for some technologies. For example, K8sGPT received a 'trial' position on the radar. By comparing across all four factors, we can see that those familiar with K8sGPT were likely to recommend it and provided reasonably high ratings on usefulness. K8sGPT also saw a higher usage rating than many other technologies but performed poorly on maturity, the worst of all technologies in the 'trial' position.

Distribution of normalised score for Observability Technologies



Legend: **Dark Green:** Adopt, **Light Green:** Trial, **Yellow:** Assess, **Orange:** Hold
Normalised score of usage, maturity, usefulness, and NPS (n=273)

PERSPECTIVES OF CLOUD-NATIVE EXPLORERS

04

4. Perspectives of Cloud-Native Explorers

Developers adopt cloud-native approaches and technologies for a wide range of reasons, and they can also have widely ranging approaches to experimenting with cloud-native technologies. Respondents were asked to self-assess their behaviour around new cloud-native technologies (the full range of options available is shown in the Methodology section). 38% of respondents stated they experiment with the latest technologies at early stages of their development. Comparing the views of these respondents to those of their counterparts allows us to explore where they diverge.

> ***Cloud-native explorers: Those who experiment with new cloud native technologies, at early stages***

Challenges

Cloud-native explorers are more likely than those who do not experiment with the latest technologies early to report not having any major challenges with either API Mgmt. and DevEx (24% vs 19%) or Observability (34% vs 28%) technologies.

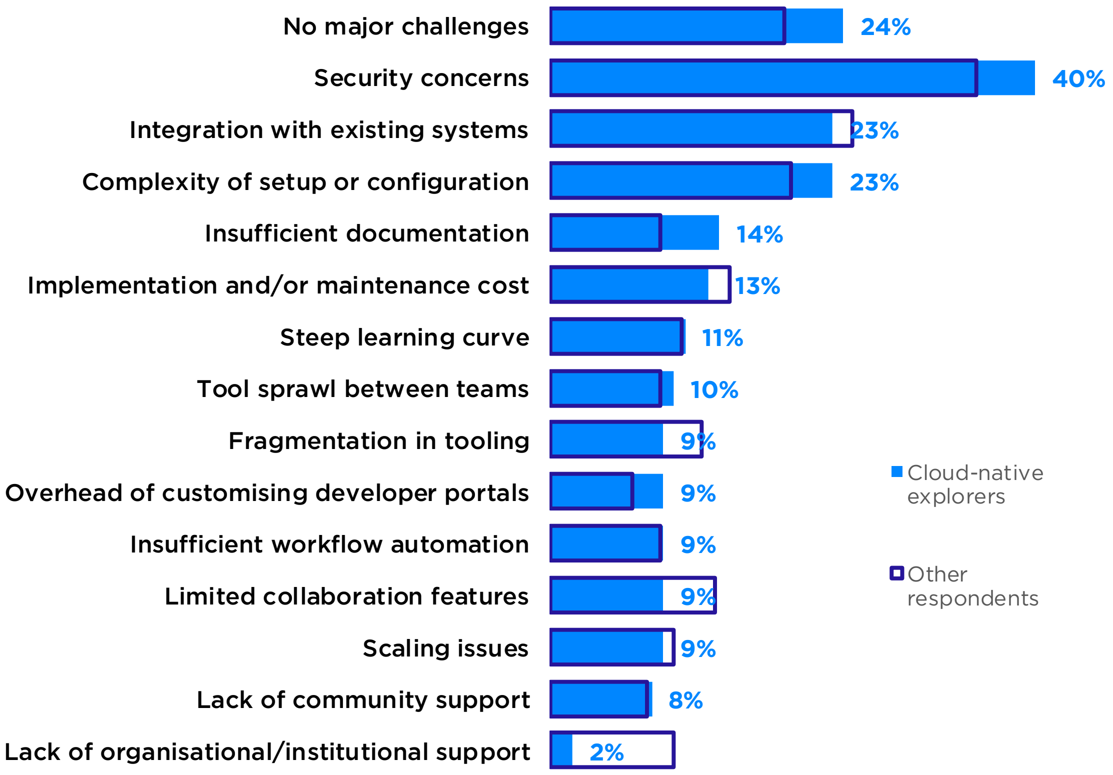
While these two groups of developers disagree on some challenges, both selected security concerns as their most common response for both technologies. However, having no major challenges was the most commonly selected option for observability technologies for both cloud-native explorers and all other respondents.

On challenges with API Mgmt. and DevEx technologies, cloud-native explores were more likely to highlight challenges with complexity of setup or configuration (23% vs 20%) and insufficient documentation (14% vs 9%) compared to other developers. On the inverse, developers who do not experiment with technologies early were significantly more likely to report struggling with a lack of organisation/institutional support than cloud-native explorers (10% vs 2%).

Cloud-native explorers were also more likely to list data latency (19% vs 11%), fragmentation in tooling (10% vs 5%), and insufficient documentation (9% vs 3%). Meanwhile, their counterparts were more likely to indicate having major issues with observability gaps (13% vs 7%), implementation and/or maintenance cost (11% vs 7%), and a lack of community support (11% vs 5%).

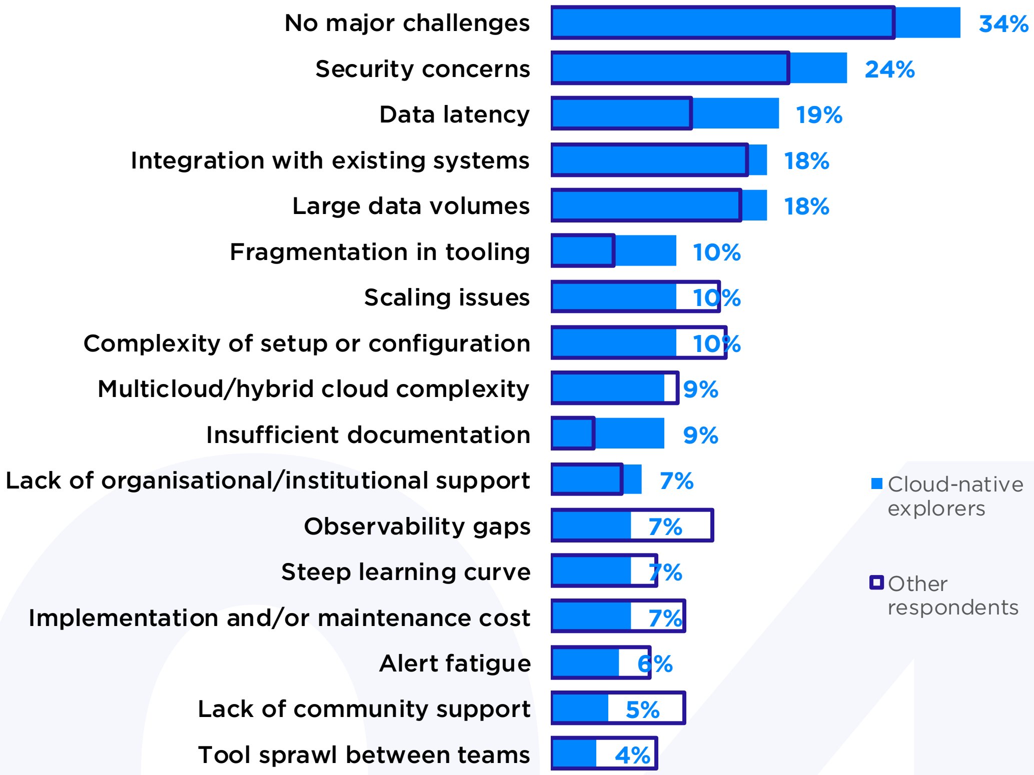
Challenges with technologies

API Mgmt. and DevEx Technology Challenges



Question wording: Do you face any challenges when using platform operation management technologies? If so, what are the major ones?
% of developers familiar with API Mgmt. and DevEx Technologies (n=280)

Observability Technology Challenges



Question wording: Do you face any challenges when using observability technologies? If so, what are the major challenges?
% of developers familiar with Observability Technologies (n=273)

Differences in Technology Scores

Diving into the scores for maturity, usefulness, and recommendation for cloud-native explorers and other respondents shows that some technologies have different perceptions from each group.

While both groups gave comparable assessments of Microcks maturity, cloud-native explorers rated it better on usefulness (78 vs 61) and their likelihood to recommend (87 vs 72) than other respondents. Cloud-native explorers rated DevSpace higher on maturity (86 vs 74) and usefulness (83 vs 71) than other respondents, but both groups had the same NPS (85).

Cloud-native explorers gave significantly larger ratings to Dapr for maturity than other respondents (82 vs 62) but were much less likely to recommend it than other respondents (72 vs 84). This further highlights that developers have a wide range of motivations behind their recommendations of technologies, outside of solely being based on maturity and usefulness. Nevertheless, these two features are strong drivers of recommendation, and both respondent groups provided high scores for all features of Backstage.

API Mgmt. and DevEx Technology Feature Score Comparisons

Category and Developer Type

API Mgmt. and DevEx Technology	Maturity		Usefulness		Recommendation (NPS)	
	Cloud Native Explorers	Other respondents	Cloud Native Explorers	Other respondents	Cloud Native Explorers	Other respondents
Backstage	🟥 82	🟩 95	⬜ 89	⬜ 95	⬜ 93	⬜ 90
Carvel	⬜ 82	🟥 75	⬜ 88	⬜ 81	🟩 91	⬜ 88
Microcks	⬜ 69	⬜ 70	🟩 78	🟥 61	🟢 87	⬜ 72
Crossplane	🟧 69	⬜ 75	⬜ 74	⬜ 76	🟢 86	⬜ 75
DevSpace	🟩 86	🟧 73	⬜ 83	🟥 71	🟩 85	🟩 85
Devfile	⬜ 79	⬜ 74	⬜ 79	🟧 73	🟩 83	⬜ 79
Score	⬜ 79	🟧 70	⬜ 79	🟥 67	🟩 82	🟩 81
Operator Framework	🟩 86	⬜ 80	⬜ 78	🟥 68	⬜ 82	🟩 84
Radius	🟩 82	🟧 69	⬜ 79	🟧 71	🟩 81	⬜ 75
kcp	🟥 61	⬜ 71	⬜ 68	🟥 59	🟢 80	🟢 80
KubeVela (Open Application Model)	⬜ 71	🟥 58	⬜ 68	🟧 62	🟢 80	🟩 76
Nocalhost	⬜ 64	🟥 56	⬜ 64	⬜ 63	🟢 78	🟩 74
Telepresence	⬜ 74	🟥 63	⬜ 77	🟧 67	⬜ 77	🟩 79
KusionStack	🟧 67	🟧 65	⬜ 74	⬜ 75	⬜ 73	🟩 80
Dapr	🟩 82	🟥 62	⬜ 78	⬜ 75	⬜ 72	🟢 84

🟥 >10pts below the average of all other cells within each row

🟧 5 – 10pts below the average of all other cells within each row

⬜ ±5pts around the average of all other cells within each row

🟩 5 – 10pts above the average of all other cells within each row

🟢 >10pts above the average of all other cells within each row

Developers familiar with API Mgmt. and DevEx tools (n=280)

4. Perspectives of Cloud-Native Explorers

Cloud-native explorers score Fluentd & FluentBit much more highly in terms of usefulness (84 vs 65) and their likelihood to recommend (95 vs 80). Inspektor Gadget sees similar patterns of greater scores for maturity (86 vs 64) and recommendation (94 vs 76), but cloud-native explorers rate as less mature than other respondents (64 vs 72).

The technology with the greatest divergence in scores is Kuberhealthy. It scores 25 points higher for maturity among cloud-native explorers than other respondents (81 vs 56) and 24 points higher for usefulness (78 vs 54). The largest difference where other respondents provided a much higher score than cloud-native explorers was for Thanos' maturity, a 19 point difference. However, cloud-native explorers rated its usefulness higher (74 vs 67) and both groups gave comparable recommendation scores.

> ***Fluentd & FluentBit performs much better among cloud native explorers***

Observability Technology Feature Scores

	Category and Developer Type					
	Maturity		Usefulness		Recommendation (NPS)	
Observability Technology	Cloud Native Explorers	Other respondents	Cloud Native Explorers	Other respondents	Cloud Native Explorers	Other respondents
Fluentd & Fluent Bit	69	65	84	65	95	80
Inspektor Gadget	64	72	86	64	94	76
Cortex	79	73	79	80	94	85
Jaeger	75	68	82	66	92	79
Headlamp	70	68	78	70	87	71
Trickster	73	68	88	68	87	69
K8sGPT	64	63	74	73	87	85
Pixie	65	75	74	75	84	79
Kuberhealthy	81	56	78	54	82	75
Thanos	71	91	74	67	77	78
Kepler	79	75	84	68	76	65
Prometheus	78	75	71	78	76	72
kube-logging	59	65	56	69	74	77
OpenTelemetry	86	76	83	72	73	80
Perses	70	71	65	69	73	75

🟥 >10pts below the average of all other cells within each row

🟧 5 – 10pts below the average of all other cells within each row

⬜ ±5pts around the average of all other cells within each row

🟩 5 – 10pts above the average of all other cells within each row

🟩 >10pts above the average of all other cells within each row

Developers familiar with Observability technologies (n=273)

METHODOLOGY

4. Perspectives of Cloud-Native Explorers

Over two weeks in February 2025, more than 300 professional developers using cloud native technologies were asked for their opinions on various observability technologies and platform tools for API management and developer experience that they were familiar with. These two technology groups were identified by CNCF as areas to investigate further for this technology landscape radar. The individual technologies selected are based on those identified by CNCF's End User Technical Advisory Board (TAB).

4. Perspectives of Cloud-Native Explorers

From the developer responses four variables were derived:

- **Weighted usage score:** Using the proportion of developers familiar with a technology, longer-term usage was weighted higher than more recent adoption, indicating long-term viability and continued use of a technology

- **Maturity score:** Developers were asked to rate technologies on a 5-star scale of maturity, where 1 star indicated low maturity and 5 stars indicated high maturity. The maturity score for each technology was calculated as the percentage of 4- and 5-star ratings, multiplied by 100 to create a scale of 0 to 100.

- **Usefulness score:** Developers were asked to rate technologies on a 5-star scale of usefulness, where 1 star indicated low usefulness and 5 stars indicated high usefulness. The usefulness score for each technology was calculated as the percentage of 4- and 5-star ratings, multiplied by 100 to create a scale of 0 to 100.

- **Net Promoter Score (NPS):** Developers were asked to state how likely they were to recommend the technology, on a scale of highly likely, likely, neutral, unlikely, highly unlikely. This was converted to an NPS by taking the percentage of likely and highly likely and subtracting the percentage of unlikely and highly unlikely. This was multiplied by 100 to create a scale of -100 to 100.

All four variables were normalised to a scale of 0 to 1, after which multidimensional scaling (MDS) was used to visualise the levels of similarity and difference between each of the technologies. MDS was chosen to provide a proximity-based analysis to identify similarly performing technologies. The results of this MDS analysis was then clustered into four distinct groups to provide the adopt, trial, assess, and hold bands of the technology radar.

Subjective nature of Likert scales

In our research, we employed Likert scales to capture developers' opinions on the maturity and usefulness, from 1 to 5-stars, of the various multicluster application management and batch computing technologies surveyed. While these ratings are inherently subjective, reflecting individual perceptions and experiences, they provide valuable insights into the developer community's views. The nature of our research is centred on investigating developer **perceptions** of these aspects, making the subjective nature of the ratings not only acceptable but also valuable for our analysis. Although the subjective nature of Likert scales may influence the interpretation of results, as different respondents may have varying standards for rating, this variability enriches our understanding of the developer experience.

Despite these nuances, analysing the distribution of ratings — such as the difference between the number of 5-star ratings and those of 1 and 2 stars — serves as a practical measure for understanding developer sentiments. This approach allows us to identify trends and patterns that can inform decision-making, highlighting areas of strength and opportunities for improvement within the surveyed technologies. Thus, we assert that Likert scales are an effective tool for gauging developer perceptions and experiences.

Respondent demographics

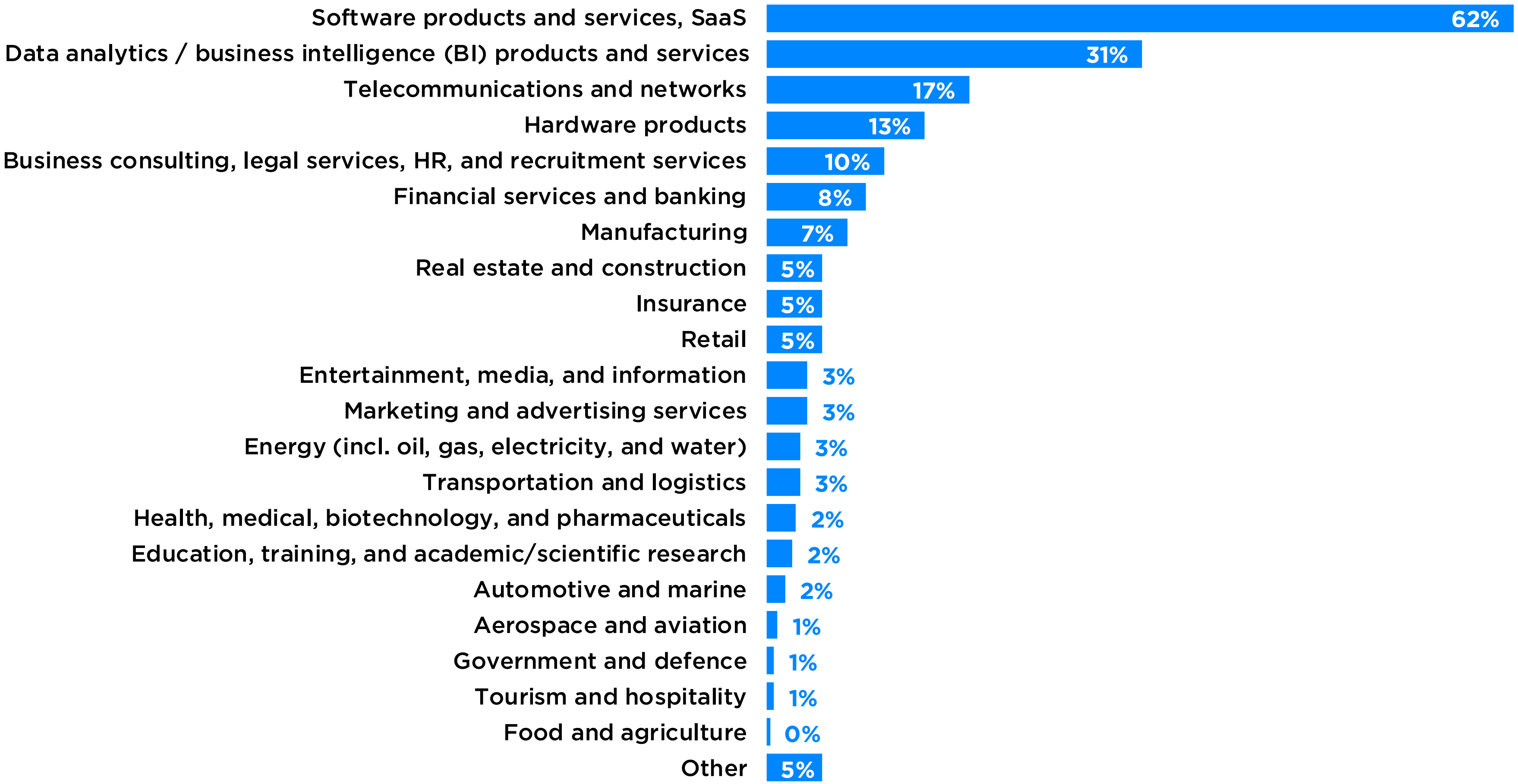
Respondents were initially asked about where their projects ran or were deployed, to identify their position as a ‘cloud developer’. Following this, they were asked which technologies they were currently using, that we associate with cloud-native development approaches, including technologies such as Infrastructure as Code, service meshes, and serverless computing.

Respondents were recruited from a combination of third-party panels, and within CNCF’s community. For privacy and data minimisation purposes, exclusion is based on internal consistency and survey-taking behaviour metrics. As such, information on the organisation the respondent works for is not carried through to any analysis. This privacy also helps encourage greater honesty from respondents, who do not have concerns that their expressed opinion will be associated with them.

Due to the nature of third-party panels making up the significant majority of respondents, we consider the risk of multiple respondents from the same organisation responding to be low, and as such do not engage in deduping cleanses. However, should more than one individual from the same organisation respond to the survey, we do not consider it to impact the validity of the results.

Within the same organisation, developers may be using different technologies. Further, while usage was used in the determination of each technology’s position on the technology landscape radar, the developer’s personal perceptions corresponded to 75% of the score the technology received.

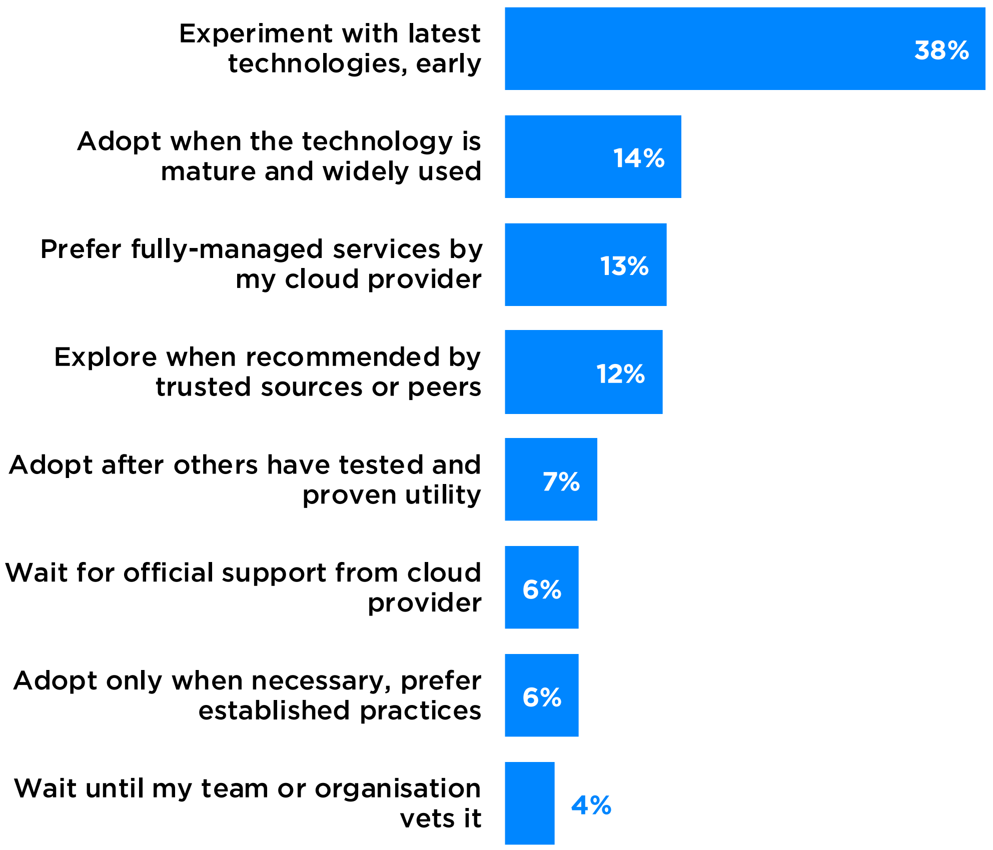
Industry involvement



Question wording: In which of the following sectors is your company active?
% of respondents (n=329)

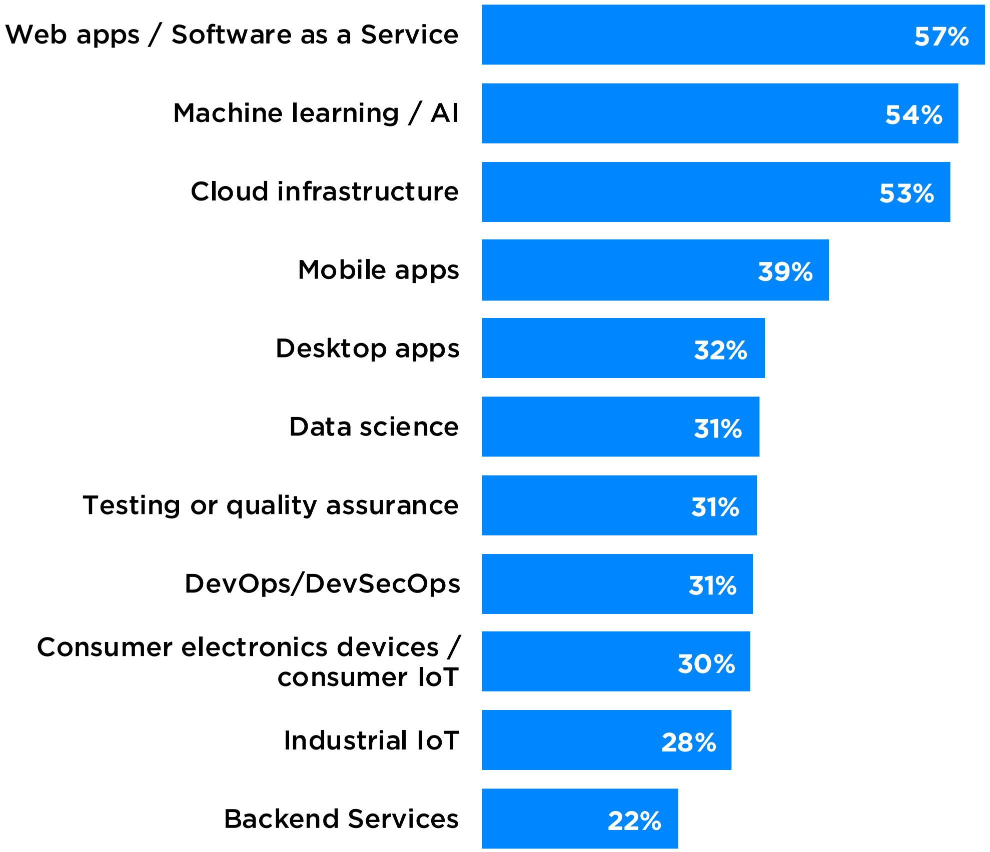
Methodology

Approaches to new cloud-native technologies



Question wording: Which of the following best describes your approach to new technologies in the cloud-native space?
% of developers (n=329)

Types of projects working on professionally



Question wording: Which of the following types of development projects are you involved in as a **professional**?
% of developers (n=329)

/DATA

Understand developers. Inspire the future of technology.

We survey 30,000+ developers annually – across Web, Desktop, Cloud, Mobile, Industrial IoT, AR/VR, Machine Learning and Data Science, Games, Consumer Electronics and Apps/Extensions for 3rd party ecosystems - to help companies understand who developers are, what they buy and where they are going next.

WHO DEVELOPERS ARE

Developer population sizing
Developer segmentation

WHAT THEY BUY

Why developers are adopting competitor products – and how you can fix that

WHERE THEY ARE GOING

Emerging platforms – augmented & virtual reality, machine learning

https://www.slashdata.co

ADATA

CLOUD NATIVE
COMPUTING FOUNDATION