Cloud Native Observability: hurdles remain to understanding the health of systems

CLOUD NATIVE COMPUTING FOUNDATION

CLOUD NATIVE
COMPUTING FOUNDATION

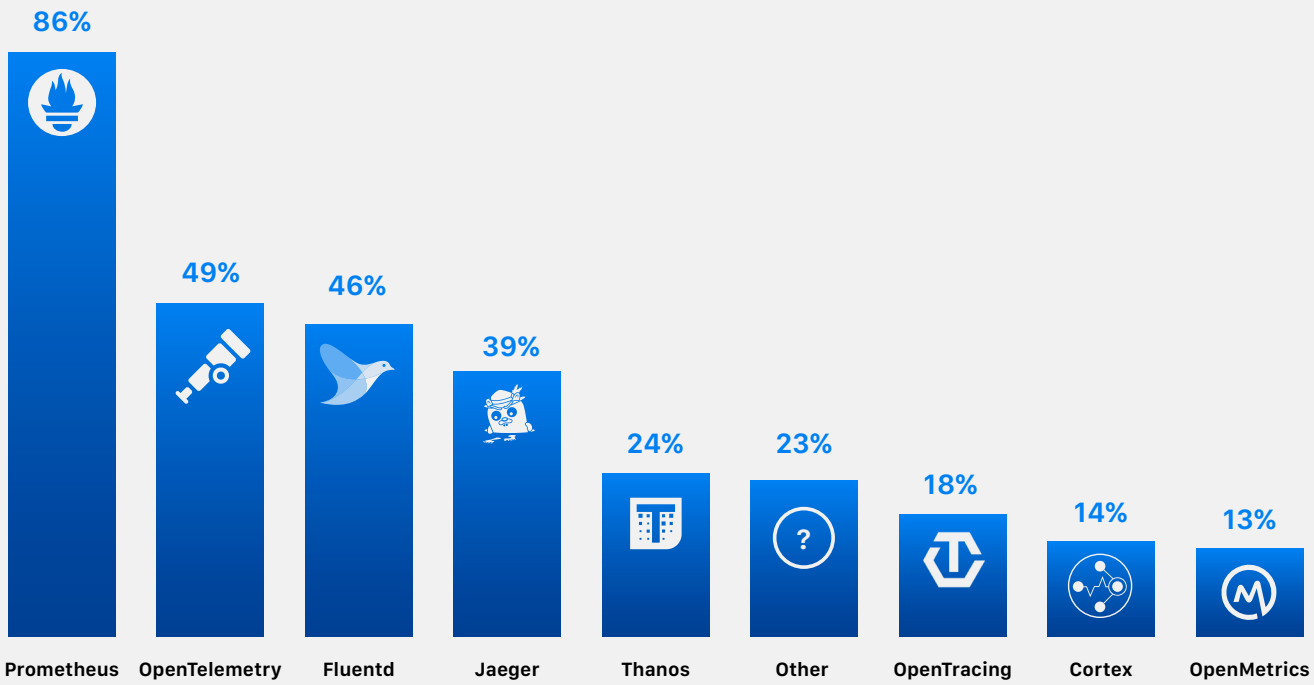
CNCF, working with the Observability Technical Advisory Group (TAG), conducted a microsurvey of the cloud native community at the end of 2021 to find out how organizations use observability tools. The survey received 186 responses.

Cloud native is the foundation of digital operations – a way to deliver agility, availability, and performance – but its increasing complexity in management and maintenance can make it hard to achieve those benefits. Cloud-based transactions are fast and operate at great scale. They involve millions of people-to-people, people-to-machine, and machine-to-machine interactions, and this environment and those transactions are producing a superabundance of data.

The desire to achieve the expected benefits of cloud has seen growing interest in a concept known as observability. This is a way to determine the health of an application, workload, or system so you can act to secure and maintain performance and availability. Observability constructed on cloud native principles, architectures, and technologies, is the best route to ensuring performance and availability.

Our survey found that three CNCF observability projects are leading the way in terms of adoption. Prometheus is the most used project, adopted by 86% of respondents and their organizations for event monitoring and alerting. OpenTelemetry, a set of tools, APIs, and SDKs to instrument, generate, collect and export data to analyze the performance and behavior of cloud native software and systems, is used by 49%. Third is Fluentd, which delivers a unified logging layer, used by 46%.

Which, if any, of the following projects do you use for observability?



CLOUD NATIVE COMPUTING FOUNDATION

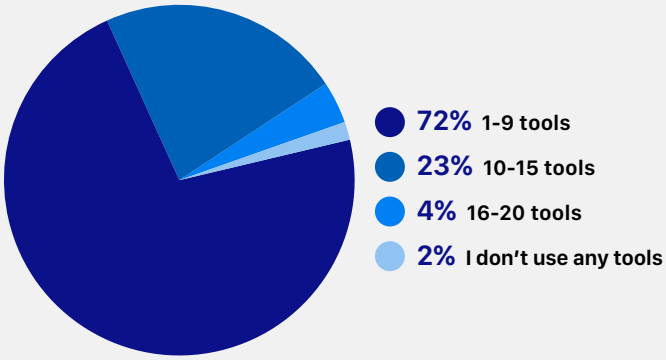
Logs, metrics and traces all help build a picture of the health of cloud systems. They target different components of the stack and are tailored for different individuals in DevOps. Some tools are open and interoperable to make data sharing easier for improved analysis and decision making.

Given the burgeoning market in observability and monitoring tools, it was no surprise that we found a huge array of logging, metrics and tracing tools deployed: 72% of respondents employ up to nine different tools, with just over a fifth using between 10 and 15.

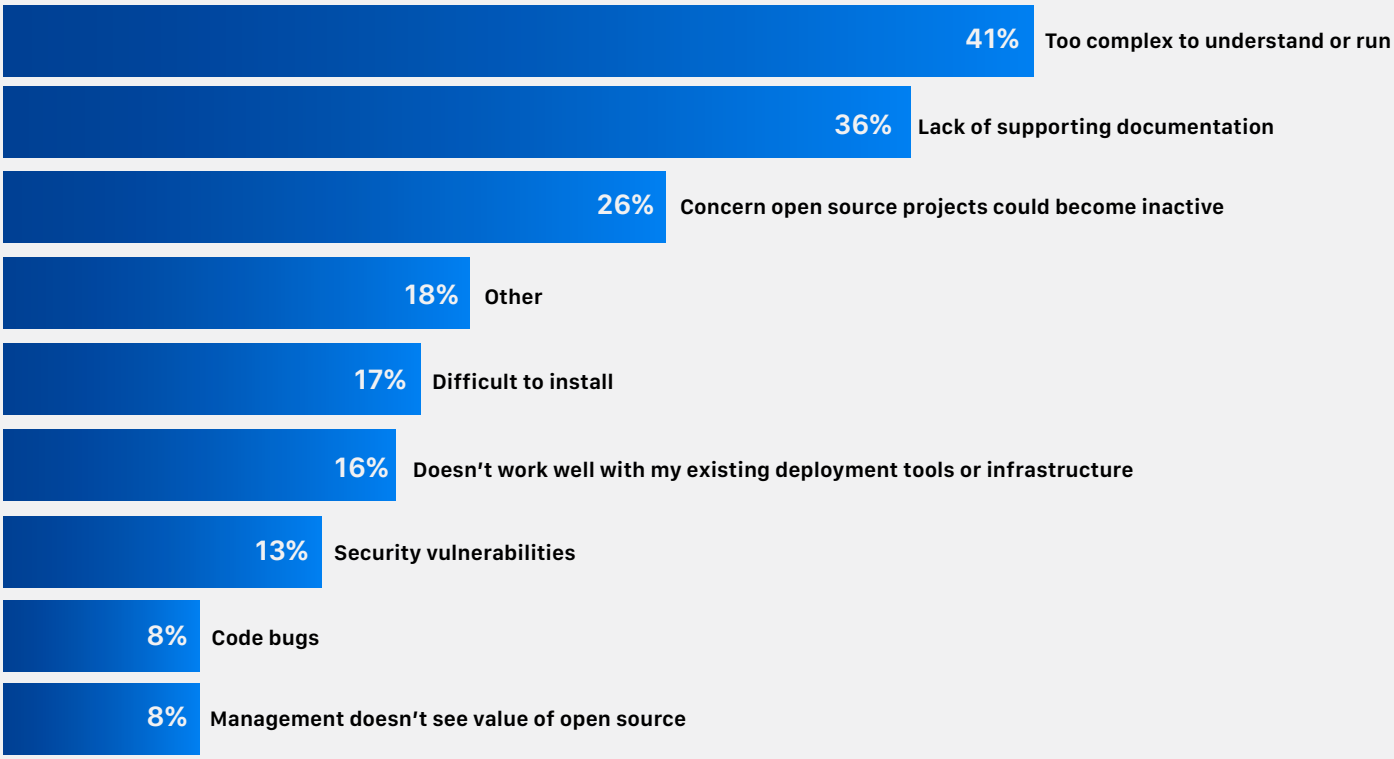
Observability challenges

We asked participants what practical, technical, or cultural challenges they had experienced, or that they foresee

How many different tools does your organization use for monitoring, gathering logging and tracing data, and for metrics?



What practical, technical, or cultural challenges have you experienced or do you foresee using these projects?



What is your greatest challenge in observability?

experiencing, in using the cloud native observability projects. The biggest challenge experienced or expected was complexity, or being too difficult to understand or run (41%). Lack of supporting documentation – materials that would help DevOps teams to build, implement, maintain and operate systems for observability using these projects as part of their long-term infrastructure – came second on the list of challenges (36%). Rounding off the top three was concern over whether projects might be abandoned or become inactive – an issue for 26% of respondents.

This latter point is a challenge all open source projects must contend with if they are to be adopted as part of core business infrastructure. Projects must demonstrate that organizations can rely on them – for example, a project development roadmap, a system of structured technical support for updates and fixes, documentation for developers, and the presence of an active community capable of sustaining each of these for the long term.

However, challenges to implementing observability weren't limited to projects and technologies, and respondents highlighted many practical, organizational issues.

The biggest challenge for half of the participants was the sheer number of tools employed by engineering teams. As seen above, most teams are using up to 15 projects. The difficulty arises when these tools lack integration and interoperability or provide a view of only certain parts of the technology stack. This perpetuates data, process, and team silos that make it difficult to develop the picture needed for a complete understanding of a systems' health. Silos around teams and processes were rated as an organizational challenge by 36%.

The continuing IT skills shortage is another problem. Lack of qualified or experienced staff has been a barrier to deploying emerging technologies elsewhere in the cloud, with projects scaled back or delayed. Lack of skills in observability is a headache for 40% of organizations.

What, if any, concerns do you have about making greater use of cloud native observability projects or products?

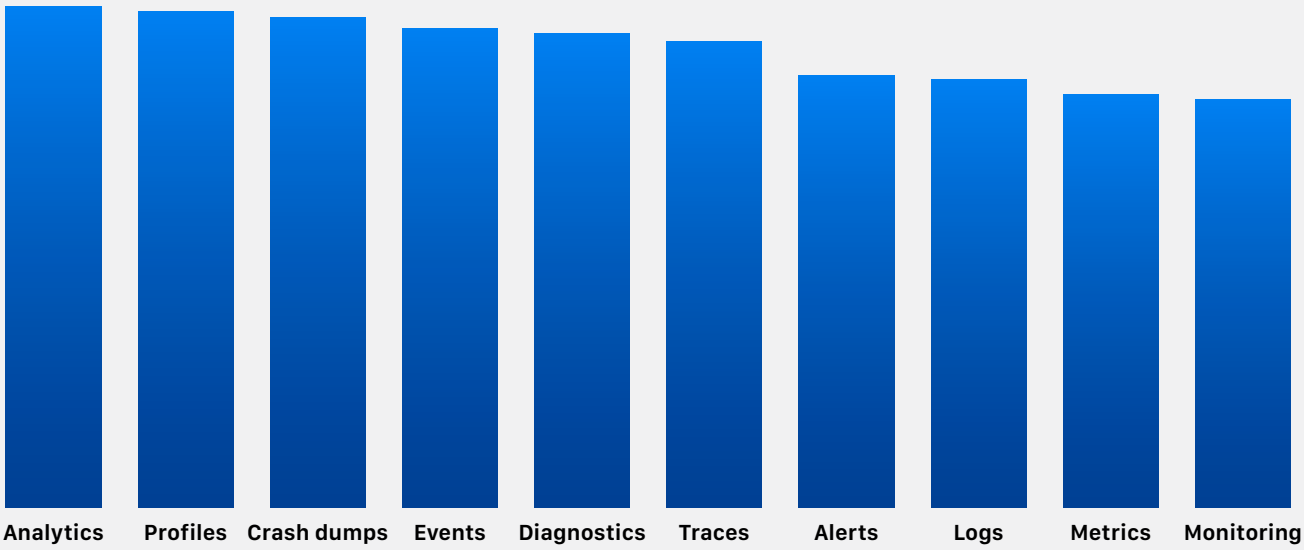
Inadequate resources were a problem for 35%. This can extend beyond the shortage of skilled or experienced staff into other strategic and tactical areas, including insufficient project budget and not having access to the technologies and guidance needed to build and provision systems adequately.

Finally, just over a third (34%) complained that their organization lacked a strategy to implement observability, while 35% said the problem was, again, complexity.

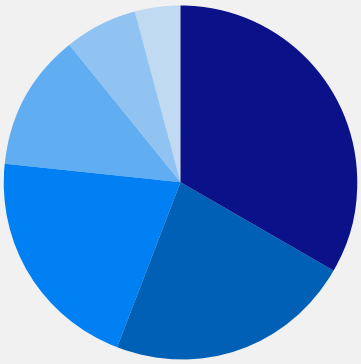
Individuals voiced several concerns about making greater use of cloud native projects and products. These concerns broke down into two broad categories. The first applied to any new technology attempting to be taken seriously in corporate environments. Concerns here centered on the ability to master new technologies, architectures, and concepts (33%). This was followed by worries about a lack of resources and expertise (31%). The viability of projects, reflected in the level of support and their long-term future, was a concern for 26%. The maturity of code, features, and products was an issue for a quarter of respondents.

The second broad category applied to all technologies, regardless of age or origin. Topping the list was integration, or the ability to work with existing and new tools and technologies (35%). Other concerns included manageability, the ability to evaluate other products on a like-for-like basis, and the ability to map dependencies and integrations. Finally, the ability of DevOps teams to optimize code was an issue for 25%.

Which of the following matter most for observability in your daily operations?



How do you run your observability tools?



- 64% Self-managed on public cloud
- 44% As-a-Service on public cloud
- 40% Self-managed on-prem
- 24% Self-managed in private cloud
- 13% As-a-Service on-prem
- 8% As-a-Service on private cloud

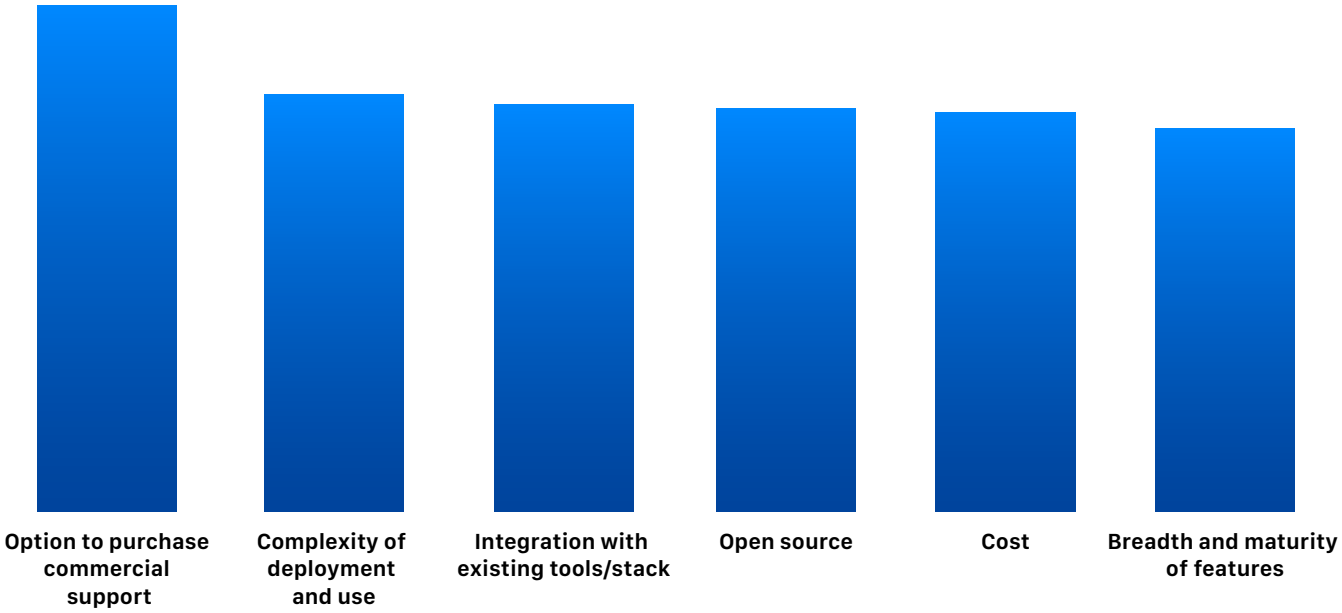
How are observability tools deployed?

There's a huge array of observability tools but how are they being deployed? The overwhelming majority (64%) are self-managed on public cloud. Observability as a service on the public cloud was employed by 44% while 40% ran self-managed on prem.

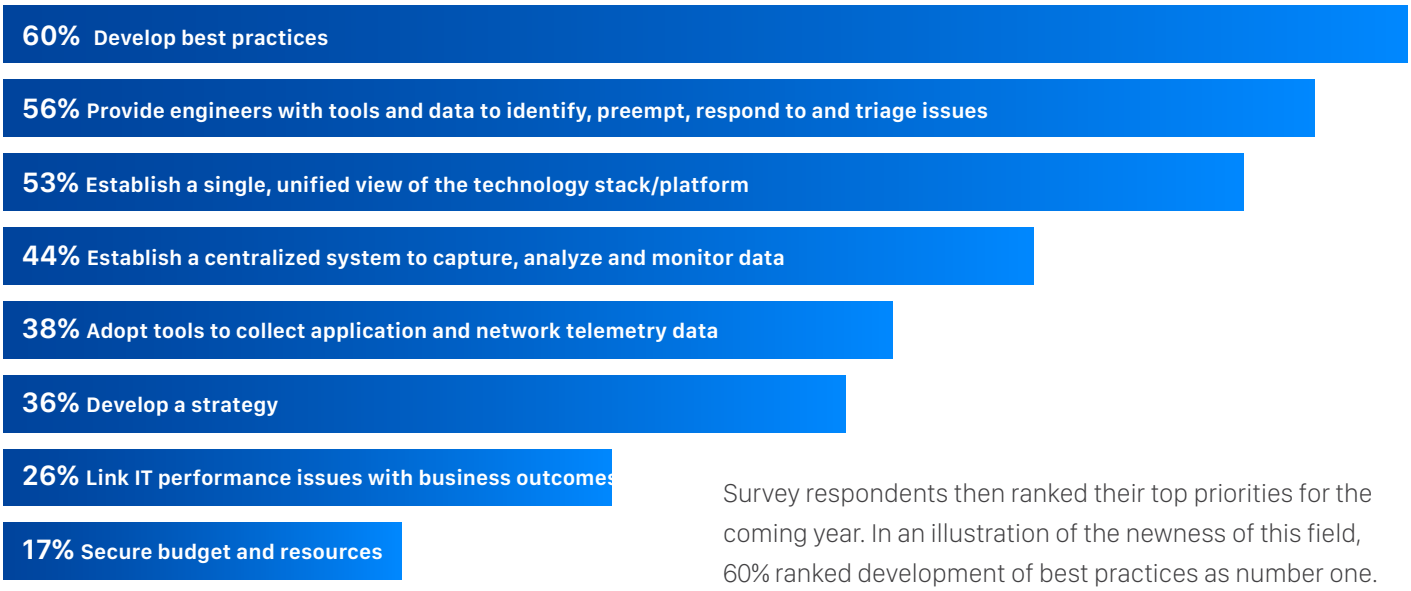
Logs, metrics and traces are considered the three pillars of observability but on their own they cannot provide a complete picture of system health. We therefore asked survey participants to rank the types of data they believed mattered for observability of their daily operations in order of importance. Analytics was considered the most important, with other data types evenly spread.

Given the challenges identified and concerns expressed, we asked survey participants to rank the criteria used for selecting observability tools. The option to purchase commercial support from cloud native projects emerged as a clear priority.

Rank the following in order of importance when selecting observability tools.



What are your top priorities or projects in observability for the next year?



Survey respondents then ranked their top priorities for the coming year. In an illustration of the newness of this field, 60% ranked development of best practices as number one. Four percentage points behind came equipping engineers with the tools and data to help them identify and solve issues to maintain system health. The third biggest priority for respondents was to establish a single, unified view of their technology stack (53%).

Methodology

The microsurvey was designed by CNCF and the CNCF Observability Technical Advisory Group (TAG). It was conducted between November and December 2021 among 186 members of the CNCF and Kubernetes communities.

Of 186 respondents:

- nearly half (49 %) were from Europe
- 22 % were from North America
- 20 % were from Asia
- the rest (eight %) were from Australia and Oceania, South and Central America and Africa.

More than half (53 %) of respondents represented enterprises with more than 1,000 employees.

- another quarter (27 %) were from organizations with 100-999 employees
- 13 % represented organizations with 10-99
- six % were from small organizations with fewer than 10 employees.

The most common job function was SIR or DevOps Engineer, specified by 39 % of respondents.

- 18 % were software architects
- 11 % were back-end developers.

40 % of the respondents worked in the software/technology industry and 20 % in financial services.

CLOUD NATIVE
COMPUTING FOUNDATION