CONTAINERS AND KUBERNETES

MARKET DYNAMICS REPORT, 2021

Efficient delivery of modern applications in any deployment environment — on-premises in a managed data centre or in a public or private cloud — is seen as a defining capability of a digitally effective organization.

This goal underpins containers and container orchestration technology and the value they provide. Containers, along with Kubernetes, an open-source container orchestration and management platform, offer a lightweight application model for quick deployment of operations based on modular, transient and immutable services.

To understand how the market is responding to containers, CCS Insight, commissioned by Red Hat, conducted a survey of more than 570 companies in Europe, the Middle East and Africa in early 2021. Respondents were drawn from a range of organizations and industries, and included business and technical roles. Survey questions examined the perceived and actual benefits of containers, their challenges and adoption patterns, the skills needed for container operations, and how containers are being used.

Key Findings

1 Adoption of cloud services is a key component of digital transformation, with organizations looking to support multiple clouds. As a result, there is a desire for a single, simplified and scalable platform that makes multicloud deployment easy.

2 Containers can address the challenges of multicloud deployments by delivering application portability. Container adoption is seen by many as a catalyst for innovation and central to progressive IT environments.

3 Simplifying container orchestration is critical to the long-term success of containers. As organizations scale their container workloads, they need tools that make management easy.

4 A shortage of skills is a barrier to container adoption. However, organizations can look to existing internal resources to support a container strategy, especially those with broad skills and experience built over several years, combined with the domain expertise of the business.
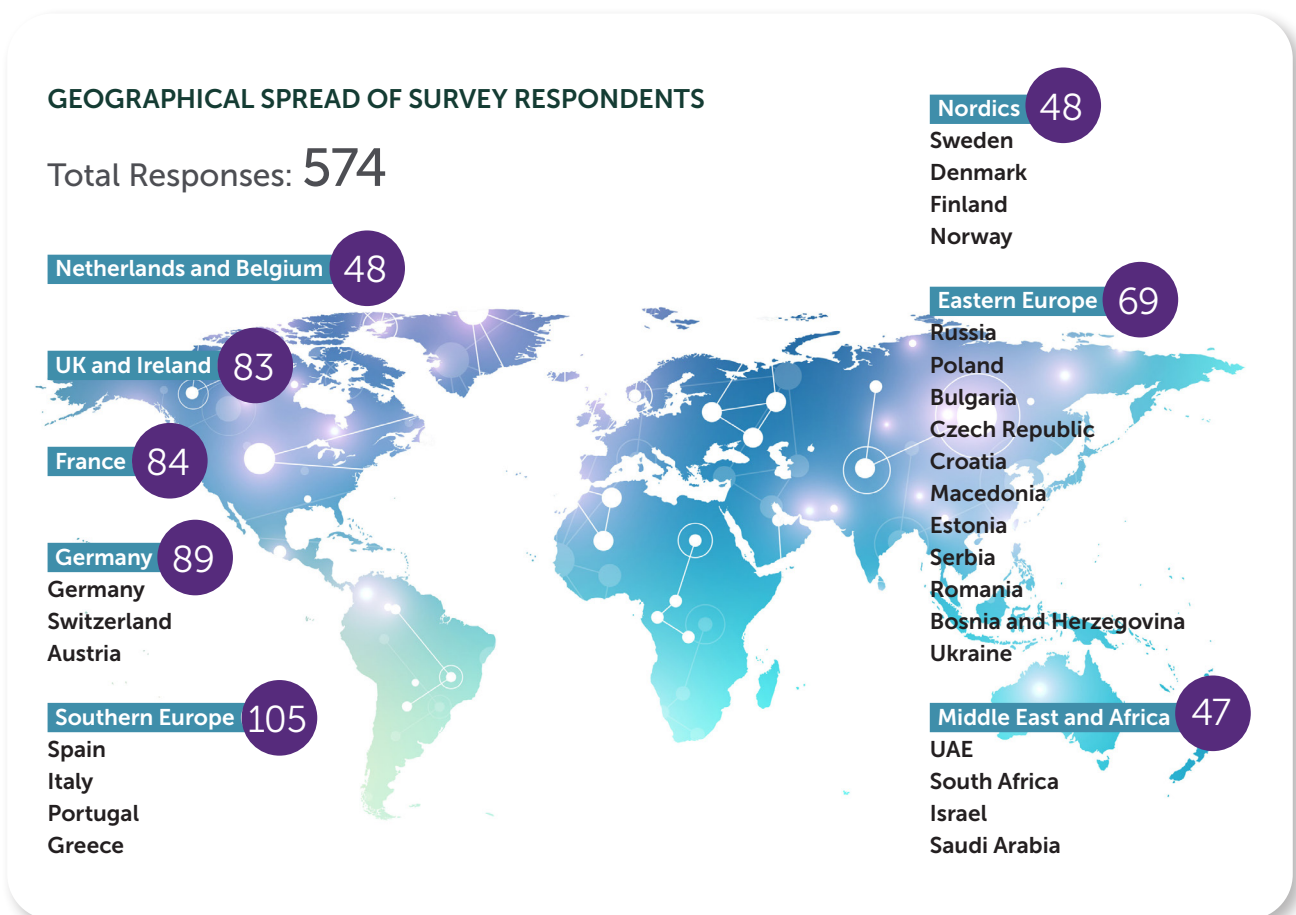
5 Containers need more than a focus on the supporting infrastructure. They present a multifaceted development platform with supporting processes that can deliver innovative and modernizing application workloads on a large scale. Attracting developers with a range of software skills and experiences is vital to the successful adoption and execution of containers.

6 Partners and suppliers will be key to successful container implementations. Survey respondents identified several attributes of a preferred supplier. Most notably trust, followed by usability, flexible pricing, support for open source, accessible training and existing relationships.

Digital Transformation Is Driving Modern Application Development

Application modernization has become the goal of many organizations. The aim is less about the app but more the processes, tools and technologies involved in its creation. It is also more focussed on building new "modern" applications than bringing existing ones up to date. The reasons for modernizing application creation are aligned with those for broader digital transformation.

The question for IT teams is how best to achieve this. There is no universal definition of which applications, supporting tools, processes and technologies to use in a modern initiative to increase innovation and speed. However, certain front runners in these different areas have emerged, one of which is containers.

Between January and March 2021, CCS Insight surveyed 574 IT professionals from a range of company sizes and industries in Europe, the Middle East and Africa. The survey sought to understand how the market is responding to containers; respondents were asked about the perceived and actual benefits, the challenges, adoption patterns, skills required and how and why containers were being deployed.

GEOGRAPHICAL SPREAD OF SURVEY RESPONDENTS

Total Responses: 574

Nordics 48
Sweden
Denmark
Finland
Norway

Netherlands and Belgium 48

Eastern Europe 69
Russia
Poland
Bulgaria
Czech Republic
Croatia
Macedonia
Estonia
Serbia
Romania
Bosnia and Herzegovina
Ukraine

UK and Ireland 83

France 84

Germany 89
Germany
Switzerland
Austria

Southern Europe 105
Spain
Italy
Portugal
Greece

Middle East and Africa 47
UAE
South Africa
Israel
Saudi Arabia

Respondents came from a mix of technical (78%) and business roles (22%), and had a broad range of experience, with 92% of respondents in technical roles having three or more years' experience of software coding or deployment.

SURVEY RESPONDENTS' EXPERIENCE AND FOCUS

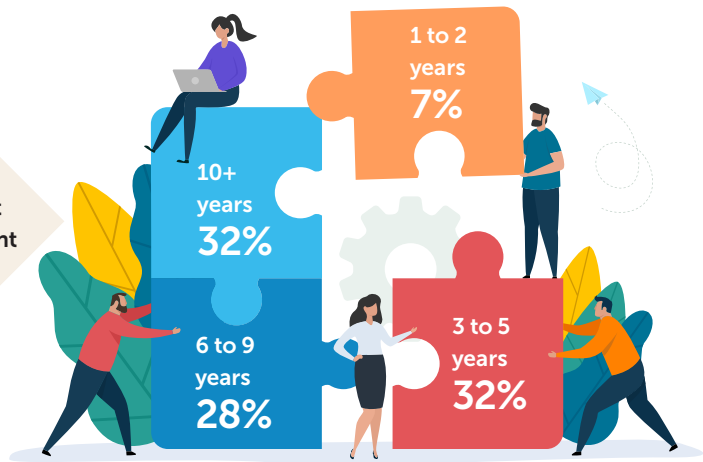**22%** Non-technical roles

78% Technical roles

83% of technical roles involved in application development

n = 574

Software code development or deployment experience

n = 373

1 to 2 years
7%

10+ years
32%

6 to 9 years
28%

3 to 5 years
32%

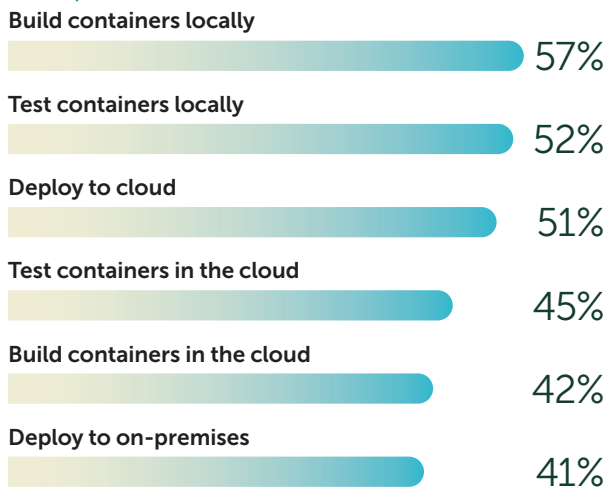# Cloud Is the Destination for Modern Workloads

Application development teams have moved from traditional "waterfall" development, with its typically long phases of requirement gathering and software development, to the faster and more iterative agile approach. Into this journey have crept additional elements such as user experience and DevOps, a natural extension of agile development that incorporates operations so that the increased development cadence can be mirrored by faster

and more regular deployments. These changes form the backdrop for the next stage in the application development journey: cloud operations.

For many organizations, the cloud is no longer new. In the illustration below, we see cloud's dominance of the tools and environments used by developers when engaged in container- and non-container-based operations.
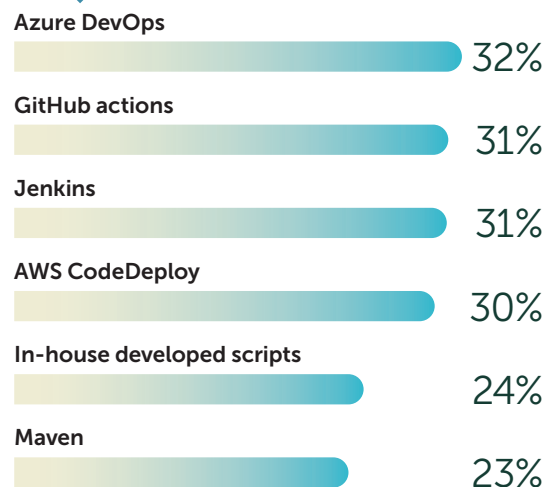
CLOUD USE IN DEVELOPMENT AND DEPLOYMENT PROCESSES

Which of the following are a part of your container development and deployment process? (Top six responses)

Build containers locally
57%

Test containers locally
52%

Deploy to cloud
51%

Test containers in the cloud
45%

Build containers in the cloud
42%

Deploy to on-premises
41%

n = 345

Which of these continuous integration, test and deployment capabilities do you use frequently? (Top six responses)

Azure DevOps
32%

GitHub actions
31%

Jenkins
31%

AWS CodeDeploy
30%

In-house developed scripts
24%

Maven
23%

n = 373

Most companies have embarked on the cloud stage of the journey ,whether through the use of software-as-a-service applications such as Microsoft 365 or Salesforce, or by using cloud infrastructure to run workloads. The question now is how to use cloud operations in a way that realizes greater potential and value.

In addition, organizations must address the complexities and variety of cloud offerings, and some must also meet regulatory needs. Most know that the answers lie in using multiple clouds and the ability to either move workloads between them or to adopt development and deployment patterns that work across clouds.

Containers As a Possible Solution to Hybrid and Multicloud

A potential solution to these challenges is the use of containers. Not a new concept, containers have been given new life in recent years. First with the emergence of Docker, initially as it was embraced by start-ups, and more recently Kubernetes to address the orchestration challenges that containers such as Docker raised. With growing support from major suppliers, containers have entered the mainstream, at least in terms of a wider awareness.

Containers help organizations move to the cloud by offering a microinfrastructure on which applications can run that can then be deployed across clouds. A developer could run a distributed application in several containers and that same container architecture could run in private or public clouds. Container technology avoids the pitfalls often encountered when running virtual machines, which can require custom configurations and are not as lightweight as containers.

Containers allow the same application to move from one cloud environment to another relatively easily. This is good when application portability is a requirement and when organizations want a single deployment pattern that can be applied to different clouds.

Our survey found that respondents see containers providing both technical and business benefits that validate their role in enabling application portability and a consistent deployment model for hybrid IT operations. But most pertinently, respondents see cloud adoption as the top benefit of using containers. This is not surprising, given that containers provide an application model that delivers the same scale dynamics as those associated with cloud delivery. In fact, the predominant types of container-based application being developed or deployed in organizations are those that simplify integration and consistency for internal systems and components. This reinforces their suitability for multiple cloud deployments.

Technical Benefits of Containers

Innovation	**39%**
Modernization	**36%**
Support for hybrid IT operations	**34%**
Scalable operations	**34%**
Improved productivity	**34%**
Cloud adoption	**33%**
Portability	**32%**
n = 345	

Business Benefits of Containers

Cloud adoption	**33%**
Innovation	**31%**
Scalable operations	**30%**
Modernization	**29%**
Improved productivity	**29%**
Cost savings	**29%**
Portability	**29%**
Support for hybrid IT operations	**24%**
n = 574	

Usage of Containers

To simplify the integration or consistency of internal systems	**43%**
To autoscale solutions that have no internal scaling mechanisms	**35%**
Development and delivery of commercial solutions for external customers	**32%**
To deliver common reusable features accross the organization	**31%**
To distribute internal shared resources	**29%**
For supply chain automated services	**28%**
n = 524	

An Appetite for Containers

Our survey showed that there is a strong mandate in organizations for the use of containers, but they are primarily in the exploratory phase. Among respondents with a technical role directly involved in building and deploying container-based applications, 46% cited containers as their top priority, ahead of improved quality, performance, and security (43%), improved continuous integration and continuous deployment (32%), and improving portability (29%).

TECHNICAL RESPONDENTS' APPETITE AND REASONS FOR CONTAINER-BASED DEVELOPMENT AND DEPLOYMENT OF NEW WORKLOADS

51% HIGH PRIORITY

9% LOW PRIORITY

40% IMPORTANT

Reasons (for those selecting important or high priority)

Deliver new types of application service
30%

Career progression in your organization
26%

Better suppport the business groups in your organization
19%

Career progression outside your organization
14%

Operational demand
11%

n = 345

Our survey data shows that 91% of technical respondents are currently developing and deploying container-based applications. Although there are workloads running in production on containers, for several (60%), it is about research and development. Respondents said that they were exploring containers for a combination of corporate and personal reasons, with 26% saying that they saw container experience as good for their careers.

For organizations, containers are viewed as a catalyst for innovation — innovation is the stated number-one development and operational benefit of containers. Although there is value in the technology itself, the wider value may be in the changes they help drive. Looking at the technologies and processes that surround the use of containers, there is a pattern of modern products, services and processes, ranging from cloud operations to DevOps. Many or all of these will be important in creating modern apps that support goals such as innovation and speed.

Other perceived benefits of containers are simplifying the integration and consistency of internal systems (cited by 43% of respondents); autoscaling existing solutions (35%); development and delivery of commercially packaged solutions (32%); and a mechanism for delivering common reusable features across the organization (31%).
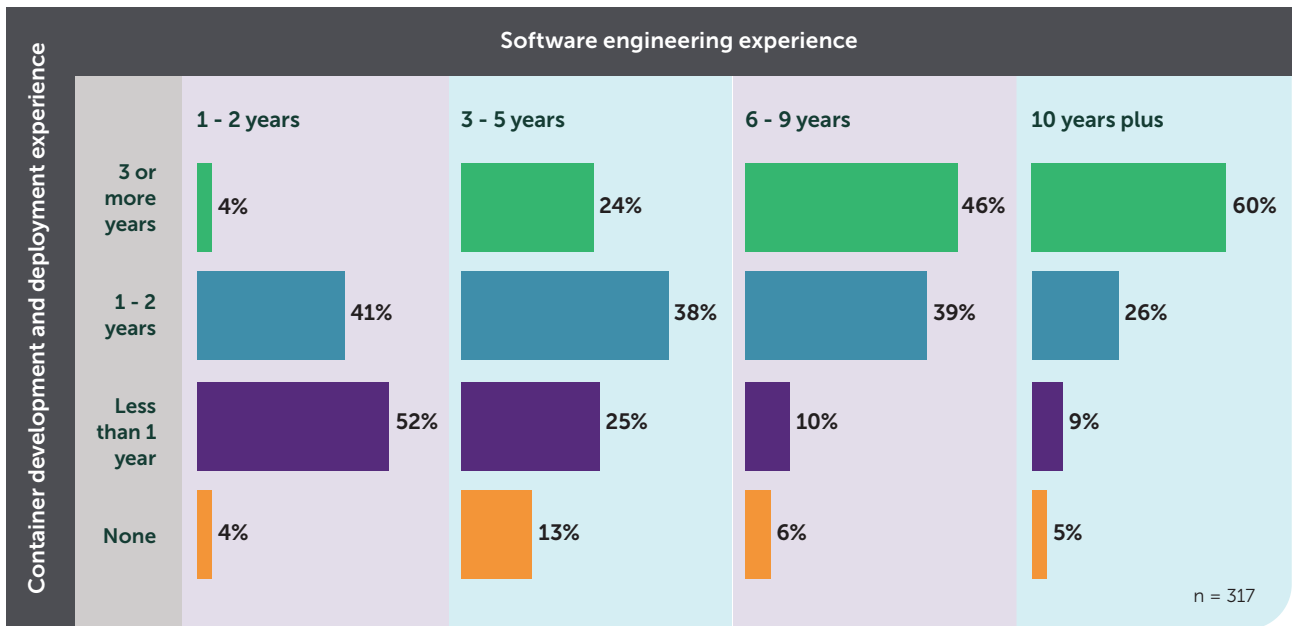
Containers play a role in acting as the technology that drives many of these others. For example, containers require skills in both the development and operations functions as they affect both. This will be facilitated by collaboration between the two. The result is that both become more aligned around DevOps than before. For some organizations that have struggled to move toward a DevOps culture, containers can be the answer.

Progressive Development Functions Are Embracing Containers

The survey revealed clearly that containers form part of a very progressive development environment, with container development and deployment experience increasing in line with respondents' software engineering experience (see charts below). Importantly, other findings from the survey indicate that the use of containers is tied to adoption of cloud-based operations.

EXPERIENCE PROFILE OF RESPONDENTS WITH CONTAINER SKILLS

		Software engineering experience			
Container development and deployment experience		**1 - 2 years**	**3 - 5 years**	**6 - 9 years**	**10 years plus**
	3 or more years	4%	24%	46%	60%
	1 - 2 years	41%	38%	39%	26%
	Less than 1 year	52%	25%	10%	9%
	None	4%	13%	6%	5%

n = 317

For survey respondents, the number-one business benefit of using containers was cloud adoption. This was especially true within hybrid cloud environments. Where multiple clouds are being used (or plan to be used) containers are seen as having a role to play. Cloud is the priority, and the leading technology and containers are playing a supporting role. But, with organizations choosing more than one cloud — private and public — they need technologies that support that choice.

A perceived benefit of cloud is its scalability, and containers are viewed as a way to better exploit that — a third of respondents cited scalable operations as a benefit of containers, reinforcing the notion that container environments can be scaled more easily and more cost effectively than traditional virtual machines.

Multicloud and Hybrid Cloud Are Key Reasons for Container Adoption

Multicloud and hybrid-cloud environments are central to understanding the appeal of containers. Especially when containers bring challenges that can lead to the question "Why don't we just do it the old way?"

The question is valid, as there are many years of experience, tools, skills and processes around traditional deployment environments such as virtual machines. Why confront the challenges of containers when these tried and tested approaches are proven — and do not preclude modern application development? After all, most current workloads in the cloud, including ones that would be considered modern, run on virtual machines.

The challenge with multicloud or hybrid-cloud environments is how to easily move applications between different infrastructures or create a single deployment strategy that supports deploying any application to any infrastructure.
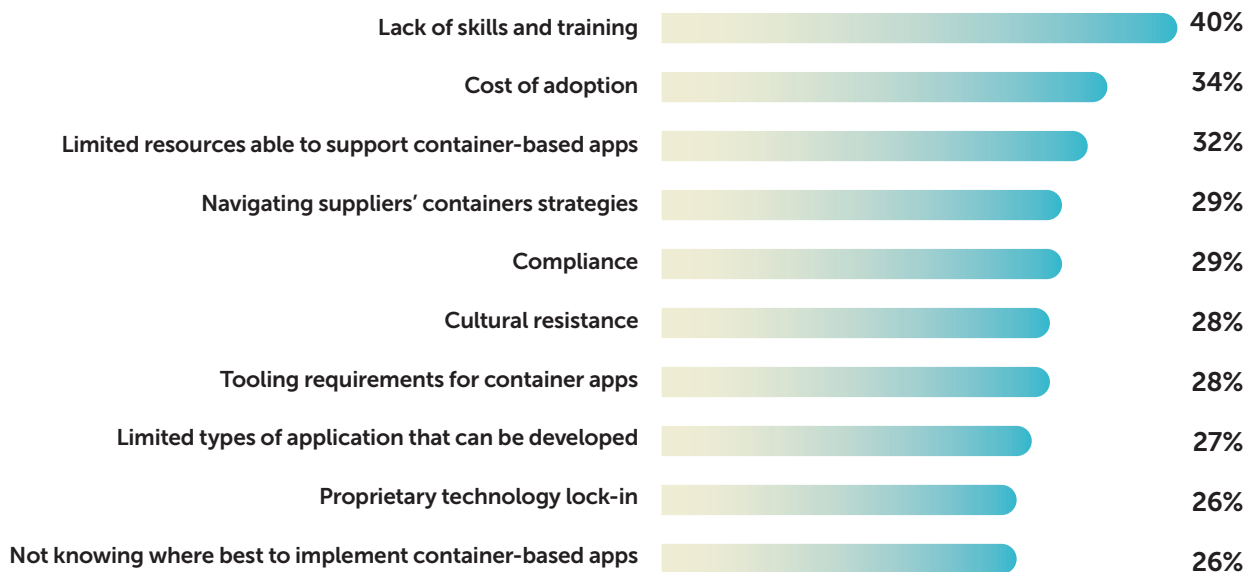
The nuance here is in the importance of portability. If the same application needs to be portable between infrastructures, then in many respects containers support this better than the alternatives. Where portability is not an issue, but organizations will be deploying different applications to (for example) different clouds, they should take a single deployment pipeline approach regardless of the target destination. Again, containers can enable this better than alternatives. This purpose of containers is reflected throughout the study results but demonstrated specifically demonstrated specifically by the cloud use in development and deployment processes charts on page 3, and the usage of containers table on page 4.
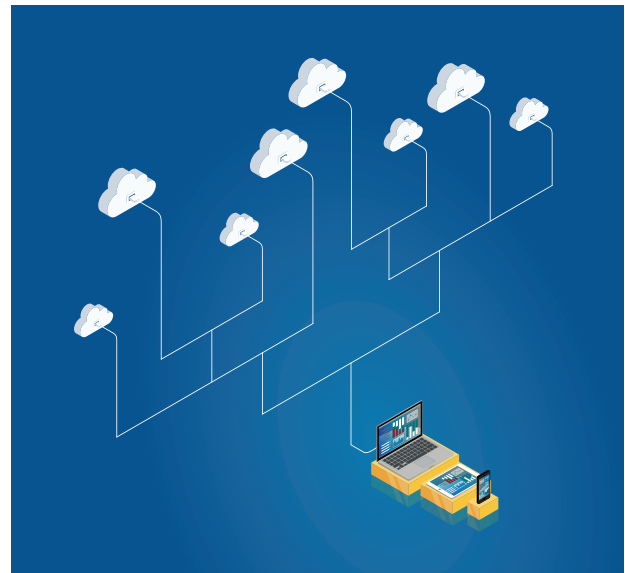
Containers Present Challenges That Need to Be Solved

Respondents identified several challenges relating to the use of containers, shown below.

TOP BARRIERS TO WIDER ADOPTION OF CONTAINERS

Lack of skills and training	40%
Cost of adoption	34%
Limited resources able to support container-based apps	32%
Navigating suppliers' containers strategies	29%
Compliance	29%
Cultural resistance	28%
Tooling requirements for container apps	28%
Limited types of application that can be developed	27%
Proprietary technology lock-in	26%
Not knowing where best to implement container-based apps	26%

n = 574

The ease with which many have come to manage virtualized environments and the ubiquity of skills are not present with containers. The study highlighted these challenges overtly but also through the profile of those currently involved with containers.

Respondents were typically skilled in multiple technologies and processes. Unlike a typical enterprise developer whose remit can be narrow (for example, write Java code), those

in the study were writing code, working with infrastructure, dealing with development and deployment pipelines and embracing a range of new tools and technologies.

Given that containers are largely in the research and development phase, this finding is not too surprising. But these individuals are not entirely representative of the wider development and operations teams in many organizations. Many enterprises with established application-building

teams will have software practitioners with development and deployment skills extending over several years. They represent a source of experience that must be tapped. However, moving containers out of the initial development phase will require a combination of new skills in a wider population, but also processes and tooling that lessen the need for such individuals.

In fact, the overriding goals of innovation and speed require individuals to be more focussed on delivering according to their skills and not spending time dealing with the challenges of containers. Developers should be developing modern applications and not worrying about container deployment pipelines.
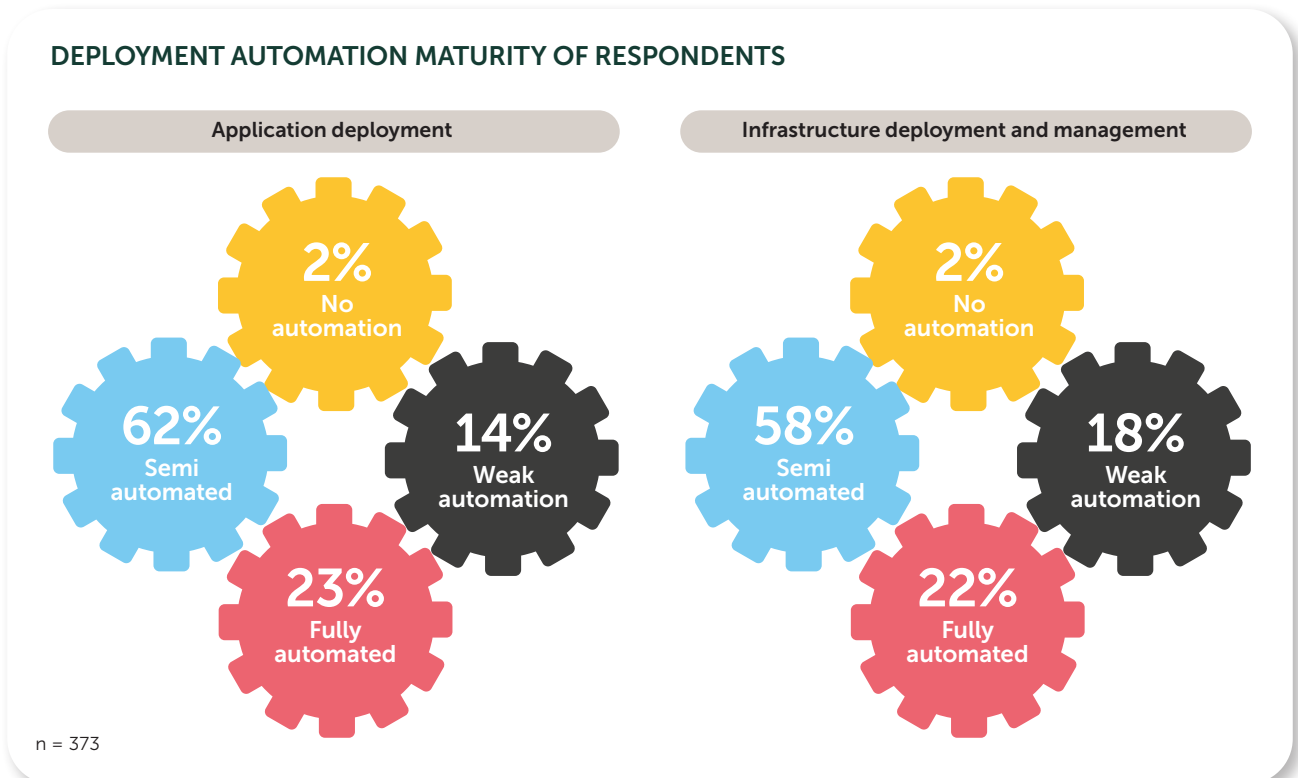
Tool Support Is a Critical Success Factor

Respondents recognize the need for tooling to support the use of containers, especially as they scale. The primary challenge with containers for some time has been orchestration. Containers tend toward highly distributed systems running within multiple containers that themselves can be replicated many times.

For example, a microservices architecture will have each service running in its own container and each container may have multiple copies for redundancy and scaling. Containers can be far more temporary than virtual machines with short lifespans. The rapid fluidity and complexity of this set-up can be challenging to manage. The rise of Kubernetes has sought to address this challenge, but Kubernetes has not been a panacea. Its widespread adoption — all major public clouds run a version of it — has helped, but container orchestration is still difficult.

Another key capability of tooling is automation. Organizations recognize that there is a growing need for automation of IT processes. Achieving scale is a challenge that cannot be addressed manually; only by automating processes will modern organizations thrive and possibly survive. The problem is not just that people can be overwhelmed, but that they cannot respond fast enough to changes; automation brings repeatability, which improves quality. The extent of automation is an indicator of the maturity and progressiveness of an organization.

We have already established the progressive capacity of the survey respondents. Therefore, it comes as no surprise to find that 85% of respondents are either fully (23%) or semi-automated (62%), as shown below.

DEPLOYMENT AUTOMATION MATURITY OF RESPONDENTS

Application deployment	Infrastructure deployment and management

2%
No automation

62%
Semi automated

14%
Weak automation

23%
Fully automated

2%
No automation

58%
Semi automated

18%
Weak automation

22%
Fully automated

n = 373

What IT teams want is a solution that makes container deployment and management both easy and automated — whether that uses Kubernetes or not. Survey respondents were clear that such orchestration is critical to the use of containers once they begin to scale. Therefore, it is key to the longer-term success of containers themselves.

It is by no means certain that just because containers are currently viewed positively as a driver of modern application development, that they survive beyond these early stages. If the challenges continue, they may be deemed too problematic to be part of a solution to optimize application development and deployment.
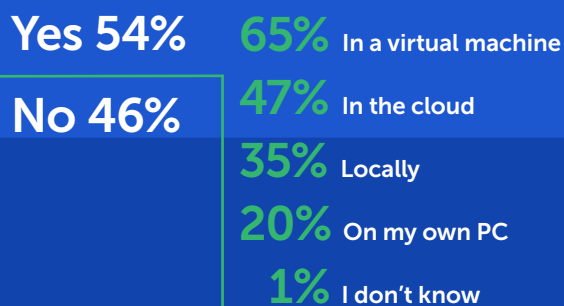
No Clear Winner Yet

The survey looked at technologies being used for containers, such as the predominant base images. Here we see that Linux is dominant even among groups that primarily develop or deploy to Windows environments. This is not surprising given that Windows container support is weak, and Microsoft's primary development platform, .NET, runs on Linux. The survey found that .NET was the third most popular language, framework or tool behind Java Spring and Python. The key considerations for choosing a container base image were security (cited by 86% of respondents), guaranteed operation (79%) and size (58%).

Respondents are building and deploying containers in a mix of locations that include on-premises environments and in the public cloud.

USAGE AND HOSTING OF KUBERNETES

Do you use Kubernetes ?

Yes 54%	**65%** In a virtual machine
No 46%	**47%** In the cloud
	35% Locally
	20% On my own PC
	1% I don't know

n = 338

Although the survey revealed a prevalence of Kubernetes, with 54% using it, it was also clear that there is no obvious winner when it comes to container support. Organizations are split about which tools and platforms are best and many will go with suppliers they know.
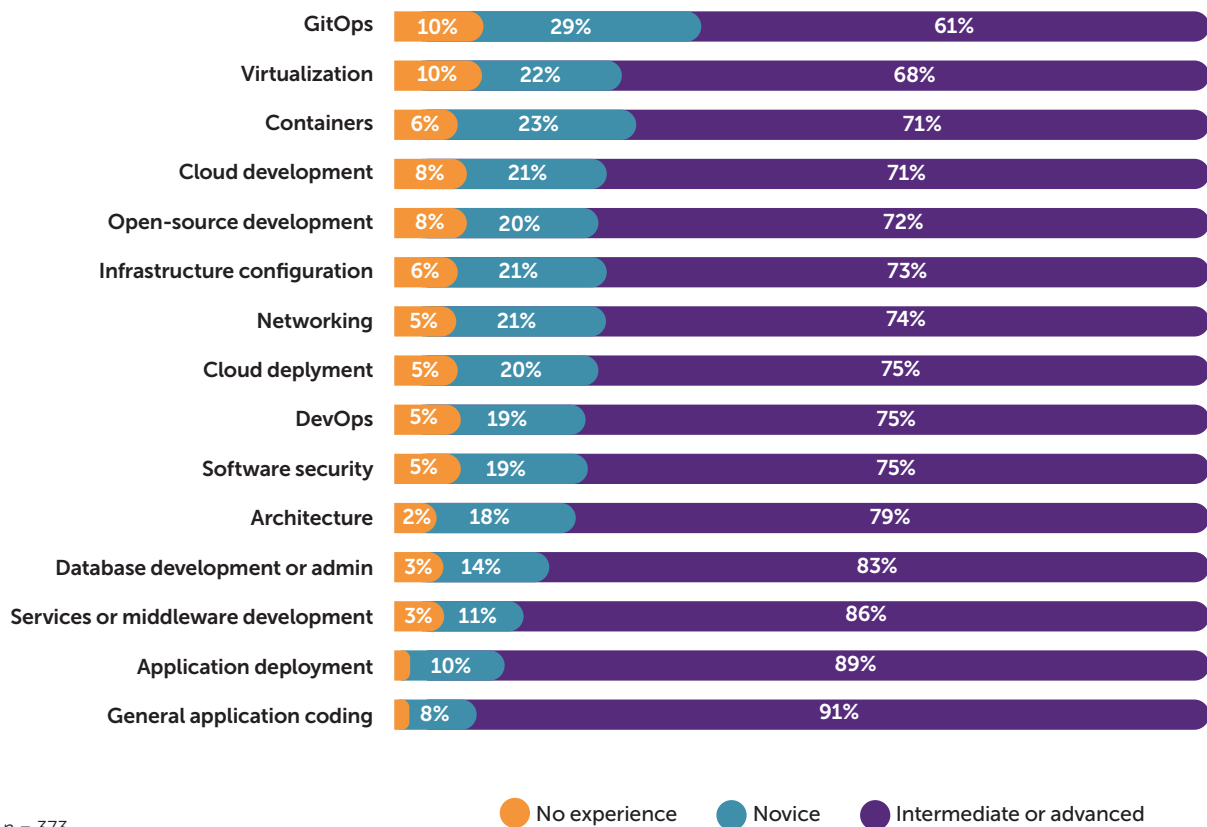
For example, in DevOps there is prevalence of Microsoft Azure (32%), GitHub (31%) and AWS (30%) proprietary tools over open-source options such as Ansible, Puppet or Chef. This may because the aim is to understand containers and these suppliers make it easier to get up and running, especially if those clouds are already being used.

The Answer to the Skills Gap Lies Within

As with any emerging technology, there is a lack of skills and this presents a challenge. Typically, an enterprise will hire in new skills and experience, but with a shortage in the market this is not so easy. Lack of skills was cited as a challenge in the survey, and this may explain the limited uptake of containers across a whole organization. Instead, they are currently the preserve of smaller groups made up of multiskilled individuals.

The survey findings shown below highlight that organizations are drawing on their more skilled and experienced IT professionals. They may not have specific experience in areas such as containers, but they have shown to be adept at working with various technologies, tools and processes.

A RESPONDENT BASE WITH PROGRESSIVE SOFTWARE MATURITY AND EXPERIENCE

Category	No experience	Novice	Intermediate or advanced
GitOps	10%	29%	61%
Virtualization	10%	22%	68%
Containers	6%	23%	71%
Cloud development	8%	21%	71%
Open-source development	8%	20%	72%
Infrastructure configuration	6%	21%	73%
Networking	5%	21%	74%
Cloud deplyment	5%	20%	75%
DevOps	5%	19%	75%
Software security	5%	19%	75%
Architecture	2%	18%	79%
Database development or admin	3%	14%	83%
Services or middleware development	3%	11%	86%
Application deployment		10%	89%
General application coding		8%	91%

n = 373

● No experience ● Novice ● Intermediate or advanced

While it appears natural to think that new developers are more likely to be skilled in modern practices and technologies, it is in fact the more experienced developers that organizations are turning to. The survey found that respondents with six to 10 years' experience exhibited the strongest capacity for container development.

These individuals are great for exploratory phases, and other organizations would do well to tap into their own experienced IT workforce. However, there will need to be a wider distribution of skills across the business. When they cannot be hired, training will be required. This will have the additional benefit of exploiting existing technical, domain and business knowledge that resides in the workforce.

Previous studies have highlighted how organizations value domain knowledge in addition to, and sometimes above, technical skills. Their goal may be modernizing application development, but, for a bank, modern applications are still banking applications, and existing staff have banking skills and knowledge.

Similarly, modern applications are likely to use traditional systems and so experience of those will still be important. There are lessons learned from the moves to web and then mobile that organizations are heeding: involve internal resources as much as possible and it is better to train your own than outsource.
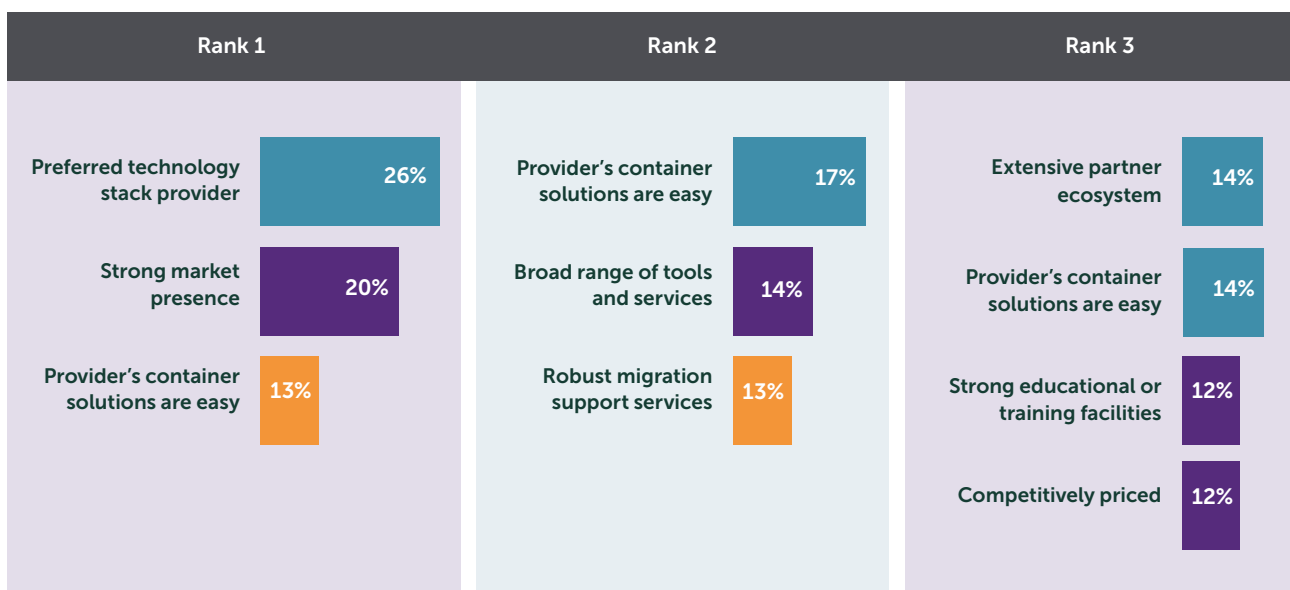
The Role of Partners

Although organizations should improve the skills of their own people, third parties still have a role to play. They can be helpful in accelerating adoption and change, but this should be tied to training internal resources at the same time. This explains the popularity of co-located projects in which the third party and the organization work in the same space (physically, and more recently virtually) to exchange ideas, knowledge and skills. Third-party suppliers are also important as they can bring platforms and tools to address the challenges associated with containers.
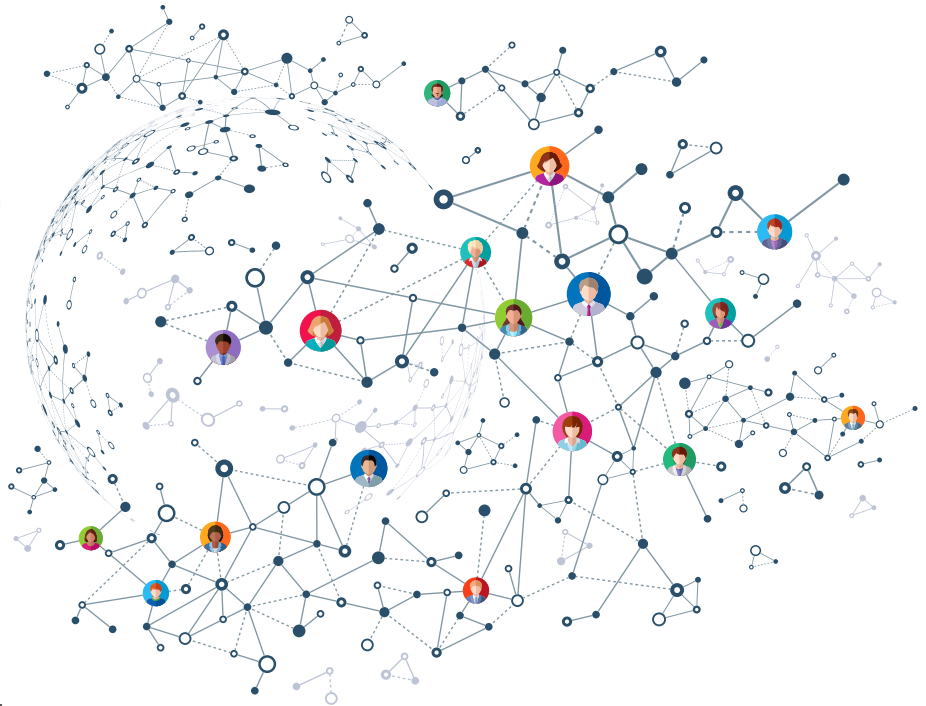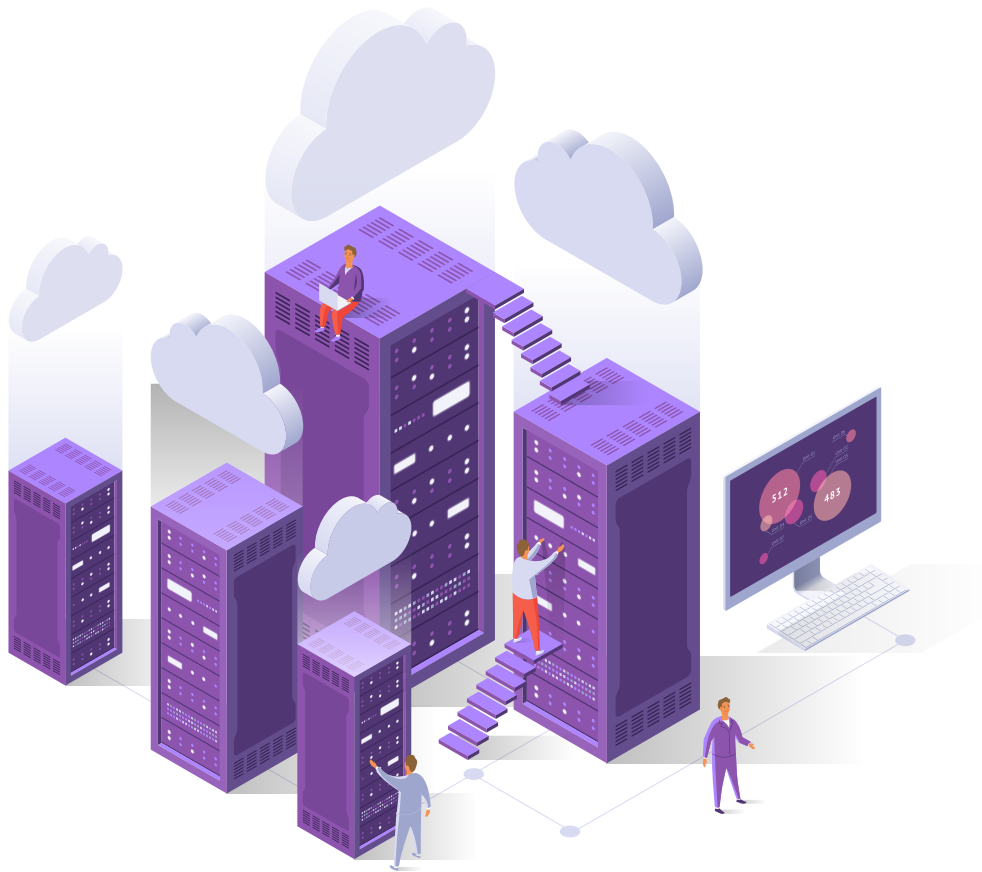
The survey found that when searching for a partner for container-based applications, 44% of respondents said trust was among their top three criteria. This was followed by considerations such as usability (cited among the top three by 37% of respondents), flexible pricing (36%), support for open-source software (31%), accessible training (31%), and existing relationships (29%).

When asked about suppliers for container development, deployment and orchestration, respondents gave priority to easy-to-use solutions, and partners with extensive ecosystems, as shown below.

TOP REASONS FOR CHOOSING PARTNERS FOR CONTAINER SUPPORT

Rank 1	Rank 2	Rank 3
Preferred technology stack provider — 26%	Provider's container solutions are easy — 17%	Extensive partner ecosystem — 14%
Strong market presence — 20%	Broad range of tools and services — 14%	Provider's container solutions are easy — 14%
Provider's container solutions are easy — 13%	Robust migration support services — 13%	Strong educational or training facilities — 12%
		Competitively priced — 12%

n = 490

Conclusion

Organizations are being driven by digital transformation and principles such as innovation and speed, and this trend has been accelerated by the pandemic. For software development teams, this means modernizing the technologies, tools and processes they use to develop modern applications. The primary focus is on the move to the cloud, and most likely hybrid and multicloud environments.

Containers have been identified as a catalyst for this broad change. They can provide a consistent platform across the various deployment infrastructures that can facilitate new processes such as DevOps. They can also address challenges such as application portability and consistency within software development or deployment pipelines.

However, containers come with challenges that include complexity, a shortage of skills, and possible costs. Complexity can be addressed through tools that include orchestration. Skills shortages are potentially best solved through training an existing workforce. One such approach would be to establish an IT environment that emphasizes the cross-pollination of software development, deployment and operational experiences.

CCS Insight's survey of over 570 technical and non-technical professionals found that containers are central to progressive IT environments that draw upon existing experienced people.

If the challenges can be overcome, containers may well have an important role to play in a future where modern applications are deployed to the cloud.

AUTHORS: BOLA ROTIBI, Research Director, Software Development and Delivery **bola.rotibi@ccsinsight.com**
CLIVE HOWARD, Associate Analyst **clive.howard@ccsinsight.com**
MARIA BELL, Associate Analyst **maria.bell@ccsinsight.com**

www.ccsinsight.com   @ccsinsight