

HOW TO HIRE FOR THE MULTI-CLOUD AND AVOID A SKILL SHORTAGE



Hiring for any position is always stressful because so much is at stake, and technical roles are particularly challenging given high applicant numbers for many titles, often-optimistic proficiency lists on resumes, and the need for careful testing for required skills. The end result is a process that can be time consuming, expensive, and even risky: the methodologies for performing this testing are hard to standardize, so a great deal of subjectivity is often introduced into what is meant to be an objective, merit-based process.

At first glance, the term "multi-cloud" doesn't seem to add much here: it is simply the act of using more than one cloud vendor. However, the reasons for doing so are usually tied to business needs that require more sophistication than just managing a few servers and load balancers in one cloud. To be successful, employees entering these teams usually require specialized knowledge and experience.

There are typically several roles found within multi-cloud teams: cloud architects and engineers, [DevOps](#), site reliability engineers (SREs), and managers. Since no one individual can credibly wear all these hats, no one job candidate will have all of these skills. Thus, building multi-cloud teams may end up feeling like fitting pieces together to solve a puzzle. It helps to start by mapping out what the pieces are, then breaking them down into specific skill areas:



Knowledge

A candidate's skill set is an obvious place to start an evaluation, but in multi-cloud environments, skills usually considered "nice to haves" may become "must haves." For example, many technical interviews focus on programming languages or vendor offerings and touch only briefly on configuration management or monitoring tools. The best multi-cloud teams make these skill areas high priorities.

Cloud and DevOps processes and solutions

Moving physical servers to the cloud is a "baby step:" the simplest form of cloud architecture on the road to multi-cloud. Multi-cloud environments are an order of magnitude more complex, so it is imperative that teams include skills coverage in areas such as tools and platforms (e.g. Terraform, Chef, or Kubernetes).

These skills are not always easily transferable between products, and while it is common to see "buzzword bingo" in resume skills lists, such as "Ansible, Chef, Puppet" in a single bullet, mastering these tools takes time and they work in very different ways. It is rare to find a candidate above the 7/10 mastery level in more than one or two, so interviewers should ensure they thoroughly evaluate for competency in the desired tools.

Likewise, companies seeking candidates with a specific area of mastery should avoid casting too broad a net. A common scorn among DevOps role seekers is seeing "Chef, Puppet, Ansible" in a requirements list. Experienced candidates will assume this red flag indicates either the company does not know what it wants, or has a mess on its hands.

Vendor Offerings

At the time of this writing, Amazon alone offered 110 distinct, branded services, and the number nearly triples as the additional top 3-4 vendors are added to the mix. Certifications may provide some hint of a candidate's qualifications, but these programs rarely cover more than a small subset of a vendor's offerings, and many qualified individuals never bother with them at all.

It helps to avoid over-specialization in the team's skills mix. Vendor offerings evolve quickly, so the ability to quickly master new technologies is generally more valuable than knowledge of a specific product. Also, a candidate with deep experience in just one vendor's offerings may be tempted to

choose the familiar product even if a competitor has recently introduced a better one. This does not mean product-specific knowledge is not important, simply that breadth often beats depth in multi-cloud teams.

Tooling

As mentioned above, experience with "tooling" may be so important in multi-cloud environments as to outweigh language or platform skills. When architectures span multiple cloud vendors, these tools may simply be the only feasible way to build, manage, and monitor those environments at scale. It is imperative that a candidate's background includes some experience in this area.

In many environments, low- (or zero-) cost tools such as [Terraform](#) may be sufficient, and because they are so common, many candidates may have used these products in the past. However, organizations with more sophisticated needs may choose to leverage enterprise-class solutions such as [BMC Helix Discovery](#) and/or [TrueSight Orchestration](#). Experience in smaller environments may not easily translate to enterprise-class infrastructures, and interviewers should evaluate for this aptitude during the hiring process.

Experience

Beyond the prerequisite technical skills, to succeed in multi-cloud environments candidates will need at least a good working knowledge of two or more vendors' service offerings. All of the top cloud service vendors offer similar compute, storage, and networking resources. But in areas such as "Big Data," serverless computing, IoT, and machine learning, the offerings diverge rapidly. A candidate with experience working with the same vendors the organization uses will have an advantage here.

Experience in the same industry and with companies of the same size as the hiring organization is also important. For example, nearly all large organizations must meet regulatory requirements, some (e.g. financial services or healthcare) to a great degree. It is very common for cloud architects and DevOps engineers to assist in meeting these requirements, and they may be called upon to determine (sometimes with "must not fail" accuracy) factors such as:

1. Where data is stored,
2. Where the customer or end-user resides,
3. Where the customer or end-user is located at the time of use,
4. How data is secured and securely transferred between sites and/or applications, and
5. Who has access to manage these components, and how this is done.

Hands-on experience in actual multi-cloud environments may be best evaluated by asking practical questions, such as "If Vendor B offers a given service at a lower price than Vendor A, will it actually be cheaper once cross-site bandwidth costs, management costs, and reliability factors are calculated in?" These types of questions are common, even daily, challenges for multi-cloud engineers. Even without a pricing table at hand, a candidate should be able to articulate a process for estimating this answer (which should typically start by asking more questions back about the hypothetical application and how it behaves.)

Vision

It may be said that multi-cloud architecture is a process, not a destination. Within the act of choosing to leverage more than one cloud vendor is the act of justifying this decision in some way. It certainly adds complexity and therefore risk, so there must be some return to make it worthwhile.

This area is a common struggle for technologists. Adopting a new cloud service cannot be done solely on the basis of a new feature or better price. The offering must fit the organization's multi-cloud strategy as well. Equally important, these teams may be called upon to communicate and evangelize their efforts more than others. A candidate with a vision for why multi-cloud is an important concept and how best to take advantage of it, combined with the communication skills to advocate for that strategy, may be a better fit than one with product-specific knowledge but no passion for "selling" others on the team on its use.

Execution

Once the other pieces of the puzzle are in place a multi-cloud team must still deliver on the overall vision, and this goes far beyond "the application runs." For instance, to determine if cost savings are being achieved, costs themselves must be measured. But, while all cloud vendors provide some type of cost reporting and analysis tool, no vendor today integrates their own data with that of competitors. Either a third party tool must be used or the team must develop its own.

Candidates with experience in one or more of these execution-related areas would bring additional value to a multi-cloud team:

1. Cross-vendor cost tracking, reporting, and monitoring,
2. Application build, test, and delivery processes in multi-cloud environments,
3. Cross-cloud backup, redundancy, and failover handling mechanisms,
4. Application and data security, particularly how to manage secure communications between multiple "walled gardens," and
5. Inventory management and reporting, a.k.a controlling cloud service "sprawl."

Consider making a playbook for how the team will achieve its multi-cloud goals, then aligning searches for new hires to expand the team's skills in areas that help execute against that playbook. As a corollary: if there is no playbook today, the next hire should have the skills required to make one!

Finally, finding the best candidates for any role is just the first step in the process. Employees must be motivated, empowered, and pointed in the right direction to achieve the organization's goals. Personal growth is an excellent motivator in nearly any role. Consider including in the evaluation those individuals with the required skills but mismatched titles, such as "system administrator" or "software architect". Remember, "cloud" itself is a buzzword invented by the industry to sell services in a new way. Skills and competence should be emphasized over past titles in this case, and the opportunity to grow into this new specialty may be a strong motivator for retaining a new hire longer than the industry average.