redapt

Insights

The Ultimate Guide to Building and Deploying Cloud Native Solutions On-Premises

Data is the lifeblood of today's economy. The key to tapping into that lifeblood is by investing in a modern datacenter.

More and more enterprise businesses are breaking free ...

... from the shackles of waterfall development cycles and are moving toward a culture of DevOps and the cloud.

Leading this change are cloud-native apps that promise increased innovation, greater flexibility, and an accelerated development cycle.

Through cloud native solutions, developers have the freedom to iterate and deploy several times a day. Errors can easily be rolled back and money saved as resources are only put to work as needed. This guide includes everything you need to know before transitioning to cloud native solutions for your business.

- What cloud native means
- The components of cloud native applications
- 5 Considerations before going cloud native
- How to build an on-prem or co-location cloud native solution
- Examples of cloud native success stories

What is cloud native?

"Cloud native" is a term that refers to a container-based development environment. It encompasses applications assembled and iterated upon using modern tools like microservices, Kubernetes, and DevOps.

Cloud native environments don't have to live entirely in the cloud. Hybrid cloud solutions and even on-prem can leverage a cloud native development environment.

Just this past April, in fact, <u>Dell announced its Data Center-as-a-Service (DCaaS)</u> aimed at enabling enterprises to utilize Kubernetes clusters within a familiar VMware environment (more on this later).



Benefits of cloud native applications

For enterprises, cloud native applications provide the flexibility for you to scale faster, cheaper, and better. <u>They also unlock new tools</u>, including:



A hybrid approach to data storage and computing which allows you to run workloads on-prem or in the cloud, depending on need



High-compute for IoT and edge solutions

making it possible to tap into cloud-like environments locally



Al infusion

to unlock automation and, when combined with edge solutions, manage applications without relying on a connection to the cloud

Components of cloud native applications



Microservices

<u>Microservices</u> decouple different segments of an application so teams can work independently from each other.

These small teams can iterate and upgrade a segment of the application without interrupting the performance of the app as a whole. This reduces the need for constant communication with other teams working on the app.

Microservices not only <u>accelerate your development</u>, they improve the stability and security of your applications because if one segment of an application were to fail, it wouldn't bring down the entire application with it.

Containers

Containers make it easy to increase the speed of your application deployment.

As each segment of an application is worked on, containers help you gather and organize the elements into a package that can be deployed independently.

Enterprises can use a tool like Kubernetes, which basically acts as the conductor for all the containers and microservices to further streamline work. Its job is to coordinate and automate workloads and ensure all your containers work in unison.





APIs

Application programming interface (API) is a communication protocol between a client and a server. Put simply, APIs lay the groundwork for how software components should work together and share data. They also help simplify client-side software.

Similar to programming models on the whole, APIs can be declarative or imperative, depending on their usage.

DevOps

A set of development practices aimed at coordinating between software development and IT, DevOps can streamline the way you handle the development cycle and help you create CI/CD pipelines.





Want to gauge your own technical maturity?

Check out our free Redapt Technical Maturity Framework eBook.

Download Free eBook

5 considerations before going cloud native

If you are <u>thinking about going</u> <u>cloud native</u> with their tech stacks on-prem or at a co-location, you need to consider these 5 things before moving forward:

2 Cost

Owning a cloud native tech stack provides an organization with the ability to control costs.

For larger organizations that have outgrown the public cloud in particular, controlling costs can translate into substantial savings.

Flexibility

One of the major benefits of deploying cloud native technology on-prem is that it provides you with more choices.

If you're a software business, for example, you may want to make applications available for specific customers in a more contained, in-house datacenter environment.

At the same time, you can make that same application available to the majority of your customers <u>at scale via</u> <u>the public cloud</u>.

From a business model standpoint, such flexibility can open up new lines of revenue without disrupting the core business.

3 Latency

If you need to run software a substantial distance from the nearest public cloud server, you may experience severe inefficiencies.

Keep in mind, though, that costs for cloud native tech stacks on-prem go beyond usage. There are also consumption costs, hardware costs, and operations and DevOps costs.

4 Customization

While there are a number of reasons to deploy cloud native tech stacks on-prem, customization is often near the top. Perhaps your data storage needs to be customized based on local regulations.

Or maybe your customers have specialized hardware needs. Similar to flexibility, customization can open a lot of doors for you when it comes to providing value to customers. It can also create sizable efficiencies in navigating local rules and regulations. In such situations, utilizing a cloud native tech stack on-prem can make both budgetary and logistical sense for your organization.

Additionally, edge computing can allow for the secure collection and processing of data on-site and in real time, further diminishing latency.

5 Technical Maturity

Your organization's level of <u>technical maturity</u> is perhaps the most important consideration to explore. Technical maturity is about whether you're actually ready and able to deploy and operate cloud-native tech stacks on-premises.

In other words, where are you at in the evolution of technology? Knowing this allows for not just clarity in whether you're capable of going cloud native, but whether you're in the best position to leverage new, innovative technology.

How to build an on-premises or co-location cloud native solution

Determining the Requirements of Your Cloud Native Solution

You need a firm grasp of your needs. That means answering these 3 questions:

- What are your business objectives?
- What is your internal capacity?
- What are your current workloads?

Building Your Cloud Native Solution

In general, building out your own cloud native tech stacks on-prem or at a co-location begins with understanding the challenges you face. These can be things like scalability, latency, and security.

To be more specific:

A) How much compute and storage capacity do you anticipate needing now and in the future? **B)** How much will latency be a factor in what you are trying to achieve?

C) What are your security needs?

Once these challenges are answered and understood, your design and build process can begin. This process usually involves four phases:



Sourcing and procuring

- **Building process**
- Configuration, testing, and validation



Solution design

Designing <u>datacenter infrastructure</u> is not unlike tuning an instrument in some ways. Every component, every compute node and hardware selection, should be tailored toward your specific needs.

2 Sourcing and procuring

Most cloud native on-prem solutions are multi-vendor due to the rarity of one single vender being able to handle all the pillars that go into a datacenter. Because one vendor may provide compute, another network, another storage and so on, procurement can be a challenge to oversee.

Add in factors such as lead times and its effect on build schedules, along with delivery logistics and assembling, and the entire process of sourcing and procuring can take up to a month by itself.





Building process

Once design, sourcing, and procuring are finished, it's time for assembly. With all the materials landing at the location, they need to be unboxed and integrated.

If you choose to go with a co-location, you will be paying for space, power, and cooling to the third party. And if you don't want to devote in-house engineering talent to the build, you'll be investing in either Site Reliability Engineering (SRE) or an additional vendor for the build.

Each of these options can add considerable cost to your project, but if you have reached that tipping point where going this direction is more cost-effective than continual investment in the public cloud, you should stay within budget.

4 Configuration, testing, and validation

Included in the building process, configuration, testing, and validation is the minutiae of the cloud native on-prem solution. Since many companies are not very mature when it comes to this phase of deployment, the process is often done through a partner.

At Redapt, we've built our own system for configuration, testing, and validation named Forge. This automation tool connects to your network and creates a profile indicating all the settings and specifications your server needs. Automation then takes this profile and lays it over a hundred servers.

One benefit of going with a tool like Forge is that it greatly reduces your risk of human error during configuration. There's no need for provisioners to take an all-hands-on-deck approach on tight timelines, or hiring employees to have crash carts. Forge is designed to simply plug in and get to work.

As for actual on-prem builds, such as within a dedicated footprint within your organization, additional costs to account for are power, cooling, and the actual space itself, as well as on-site personnel—or a third party vendor—to oversee the build and ongoing management.



Cloud native success stories

We've <u>previously covered five companies</u> — including the likes of Capital One and DISH Network — that have achieved a competitive advantage by going cloud native.

Here's another example:

The Gaming Company

When an iconic gaming company was looking to go cloud native, they partnered with us to help bring their IT ops team in charge of maintaining infrastructure up to speed on using containers and Kubernetes.

We took the company through every step of deploying Kubernetes

to accelerate their development process and drastically reduce potential downtimes. Now the company's team is able to utilize and manage containers and Kubernetes. In addition, downtimes that would have previously lasted for hours and even days were reduced to just 15 minutes.

Our recommendations were:

- DevOps, containers, and Kubernetes capabilities for the company's ops team.
- Infrastructure upgrades and enhancements to handle new development processes.
- Automated backups to reduce downtimes from potential days to a mere 15 minutes.

Read Full Case Study

🕝 redapt

Unlocking the Potential of Modern Data Centers

Five Good Reasons Companies Should Invest in Their Data Infrastru<u>cture</u>

<text><text><text><text><text><text>

Cost Savings Each day, more than 2.5 quintillion bytes of data are created globally.⁴ Capturing and storing this deluge requires a transferior amount of investment

Free eBook

Unlocking the Potential of Modern

By investing in modern datacenters, companies large and small can leverage the power of the cloud to save money, streamline workflow, increase productivity, maximize data capital, and fully transform into a digital enterprise.

Download

In Conclusion

Going cloud native has its benefits and its potential pitfalls.

The keys to being successful are:

- Knowing what your goals and needs are beforehand
- Understanding the costs involved in your own cloud native solution
- Assessing your technical maturity
- Investing in the capabilities or partnering with experts

For some examples of the type of equipment we often recommend, check out tech we like for <u>cloud native tech stacks</u> and <u>rack integration services</u>.

Get in touch with us today

<u>results@redapt.com</u>

(425) 882-0400

