MONTEL
INTERGALACTIC

YOUR GUIDE TO

Cloud Infrastructure Modernisation

THE ABC OF THE TOOLS, TECHNIQUES AND PLATFORMS

Welcome

Your business is growing fast and your critical services need to handle uneven load. The software architecture you have carefully built may have ended up complex and prone to outages.

Even though the current system is not perfect, things still work for the moment. The future however holds another ball game and your software infrastructure needs to be ready for it. Yet growing your core business instead of worrying about infrastructure problems is what you want to do.

In this guide, we present the best tools for your infrastructure modernisation path.

Jussi Sarkkinen
Montel Intergalactic

4 steps to an efficient, scalable & fault tolerant infrastructure

1 Containers as the Application Medium

Modern encapsulation technologies such as Docker provide a very welcome addition to application deployment.

Proper encapsulation allows developers to run identical code on their local machines and isolate errors before they crash your production system.
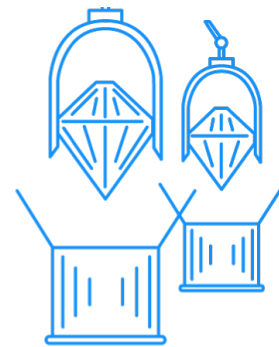
2 Cloud Migration - Pick the Best Infra for Your Needs

There are dozens of potential service providers out there. Choosing one is a delicate process that depends on the requirements posed by your service.

If you are unsure which one to choose, it is a good idea to trust professionals to pick you the most suitable one.

3 Kubernetes to Manage Your Infrastructure

Kubernetes is the de-facto technology for application orchestration in the modern cloud infrastructure. Setting it up properly and making it do your bidding is not easy but there are Certified Kubernetes Administrators to help you out.
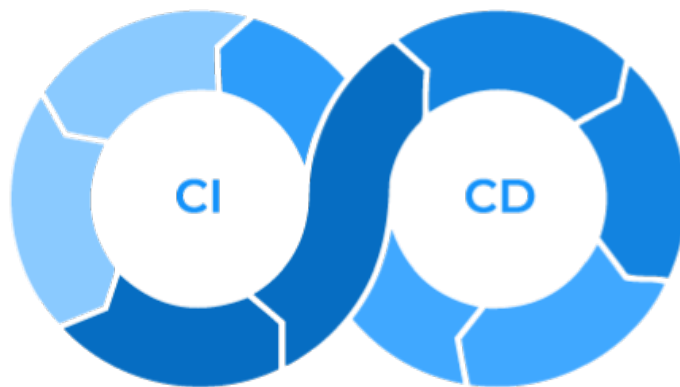
4 CI/CD Continuous Integration and Deployment

You now have a working infrastructure layer. Excellent. Then, let's focus on the application layer. A good CI/CD is a life-saver that will ensure your service will not go down due to accidental bugs that are overlooked during development.

2

Modern Application Development and Deployment Using Containers
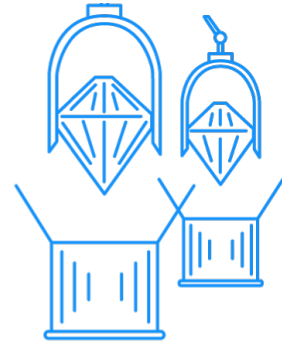
Deploy software fast and operate at a larger scale, boosting your SaaS business to a new level. Software application development and orchestration has been revolutionized with the arrival of container technologies such as Docker.

Developing, deploying, and running your applications in isolated containers gives a boost in scalability, productivity, and developer happiness.

High productivity

Containers are lightweight, isolated processes that enable more efficient use of system resources. Quick and easy setting up development and new server environments saves time in onboarding new software developers and testing new features.

There is consistency between deployment, testing, and production environments, simplifying and speeding up your deployment significantly.

Agility and flexibility

Docker is easily portable across platforms. Applications running in containers can be deployed easily in multiple different operating systems and hardware platforms.
With containers, your dev and ops teams spend less time debugging and diagnosing differences in environments, and more time shipping new functionality for users.

On your infrastructure modernizing path, investing in container technology such as Docker is almost a prerequisite. The container orchestration platforms such as Kubernetes provide even further benefits such as horizontal autoscaling.

Providing security

When things break, the whole system won't. Because the applications are isolated from the host system and each other, the damage is minimum and control is maximum. Every microservice runs in complete isolation with complete control over traffic.

Docker has also implemented default configurations that provide greater protection for applications running within the Docker Engine, making a strong security default for all containerized applications while still leaving the controls with the admin to change configurations and policies as needed.

3

Cloud Migration – Find the best solution for you
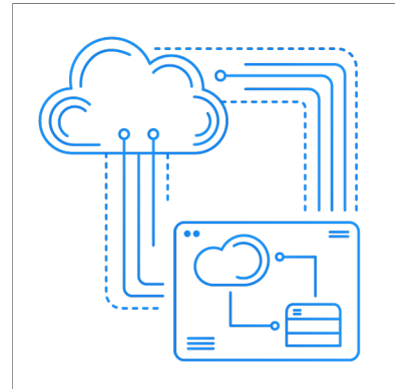
Optimizing your cloud usage requires understanding the various platforms and technologies as well as your current service architecture and the future vision. Selecting a cloud provider is a buyer's market right now with plenty of viable options available in various different price ranges. Picking the most suitable is not trivial and too often we see choices made by herd mentality.

Architecture first

Seeing the bottlenecks, growing pains, and points of failure in your service architecture helps you decide the optimal cloud provider. Thinking of the optimal solution should start with the current situation but always aim for the desired future vision. Predict your future load and customer behaviour.

Considering different scaling techniques and load-balancing has a big impact on the application-level architecture. Microservice architectures and serverless solutions provide new ways of scaling your services and serving your customers better and more reliably.

Significant savings

Using a full feature rich cloud platform such as Google Cloud, AWS or Azure might not make that much sense if you only want to run machine instances. Moving your instances to a lower tier service provider gives you significant savings in infrastructure costs and simplifies your setup considerably.

When staying with a large cloud provider, make sure you use their resources wisely. Consider running managed databases, shared volumes, lambda functions, and all the bells and whistles that are provided. Avoid running just plain machine instances.

Mix and match

Concentrating your applications under one roof has its benefits but maybe you want to keep your options open instead of heading straight to a vendor lock-in.

There are many cases where using a single provider is not optimal. Maybe you want to run your frontend on AWS with all its CDN glory and provision huge bare-metal servers for number crunching or a local service provider for sensitive materials and intra services. In these cases, interoperability between the platforms has to be ensured and kept secure.

Kubernetes as a Solution to Orchestrate Distributed Applications

Efficiency, Scalability, and Fault Tolerance

Orchestrating your application containers in clusters and running them with Kubernetes enables efficiency, scalability, and fault tolerance.

Certified Kubernetes Administrators know the ins and outs of the brilliant tool and can implement it for you with experience.

Cost efficiency

Having multiple static environments running on manually provisioned machines will leave you with a complex, hard to update, and costly infrastructure. With Kubernetes, you only need one cluster with colocated applications on the same machines, reducing the cost of hardware.

Simplicity & Scalability

Kubernetes was originally developed by Google for automated deployment, scaling, and management of containerized applications. So, Kubernetes is one tool to manage your distributed applications. This allows for a flexible architecture that integrates nicely with modern need-to-have features such as a CI/CD pipeline and auto-scaling, which will save you in infrastructure costs while improving user experience.

Reliability

With Kubernetes, you'll have high availability as applications are self-healing, distributed to multiple machines, and load balancing. You will also have zero downtime due to rolling automated updates and roll-backs as a back-up when a major incident occurs.

5 Continuous Integration and Deployment

A working ci/cd pipeline ensures that you have direct visibility to your latest development version, your production systems rarely fail and your developers are happy.

Your business is torn between two extremes: on the one hand, you need to move fast and beat your competitors with new features, but on the other, you are serving real customers who depend on your service not failing them in the middle of the day. Continuous integration and deployment is the shining beacon many development teams aim towards.
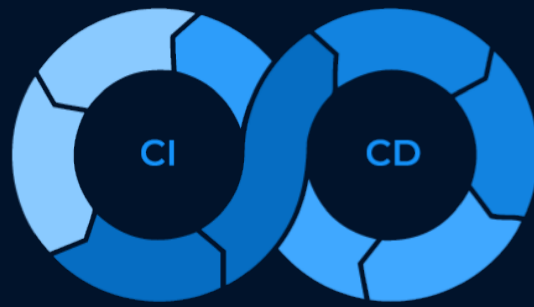
Move fast

A working CI/CD setup is a 24/7 workhorse that acts every time new code is pushed to version control. It frees your developers from running fragile local tests or (gasp) pushing their local code to production servers. Your developers can focus on the bleeding edge and trust that the pipeline will catch any regression bugs they might be causing while dashing through the codebase.

But don't break things

The I in CI/CD is responsible for running your test suite. You do have a test suite covering mission critical features, right? If not, that should be fixed soon. If the tests are not all green, nothing on the servers will change, and the people responsible for breaking the codebase will know it was them.

Empower and remove bottlenecks

An often repeating anti-pattern in development teams is that only one or few members can do new releases for production and testing purposes. A good CI/CD pipeline will allow all your developers to do production hotfixes in the middle of the night or push a new development version for QA.

Looking for an optimal cloud modernisation path for your needs?

Are you constantly facing challenges in managing your cloud infrastructure? Is your team overwhelmed and your systems prone to unexpected issues? MontelCare could is the perfect solution.

MontelCare is designed to reduce IT challenges for businesses by providing comprehensive support and expertise in cloud service management. It allows you to focus on your core business while our experts take care of your infrastructure.

Book a free consultation today!

Jussi Sarkkinen
Co-founder
+358 50 354 7200