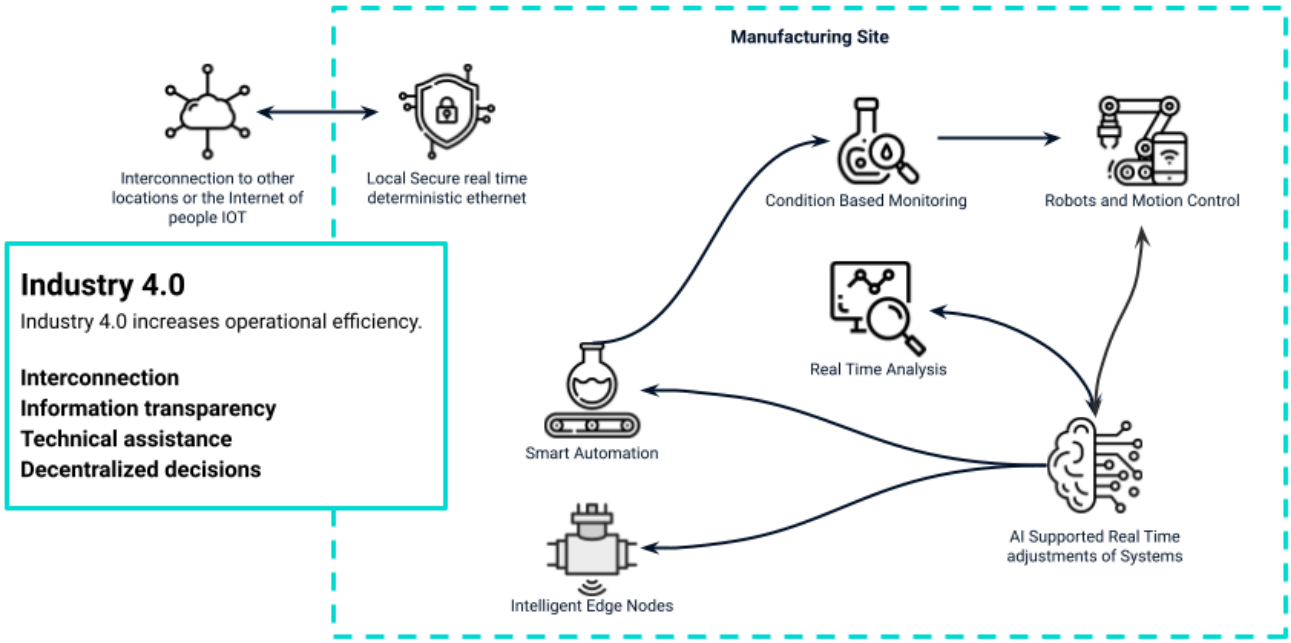
Manufacturing Edge

The history of industrial manufacturing is an interesting journey to take. It is a story of mankind enabling itself to produce machines that produce machines in the late 18th century with the help of steam, water and pressure. In the early 20th century electricity was common and mass production powered a revolution that everyone followed soon after. In the mid 20th century computers were invented and were integrated in manufacturing in what we know today as Automation.

In the present, success of Industry 4.0 or Manufacturing in general is measured in the system's degree of autonomy. For example: How many hours of unattended and uninterrupted work can an assembly line deliver, is packaging integrated and can I shift an assembly line to produce different products with the same level of efficiency and integration?

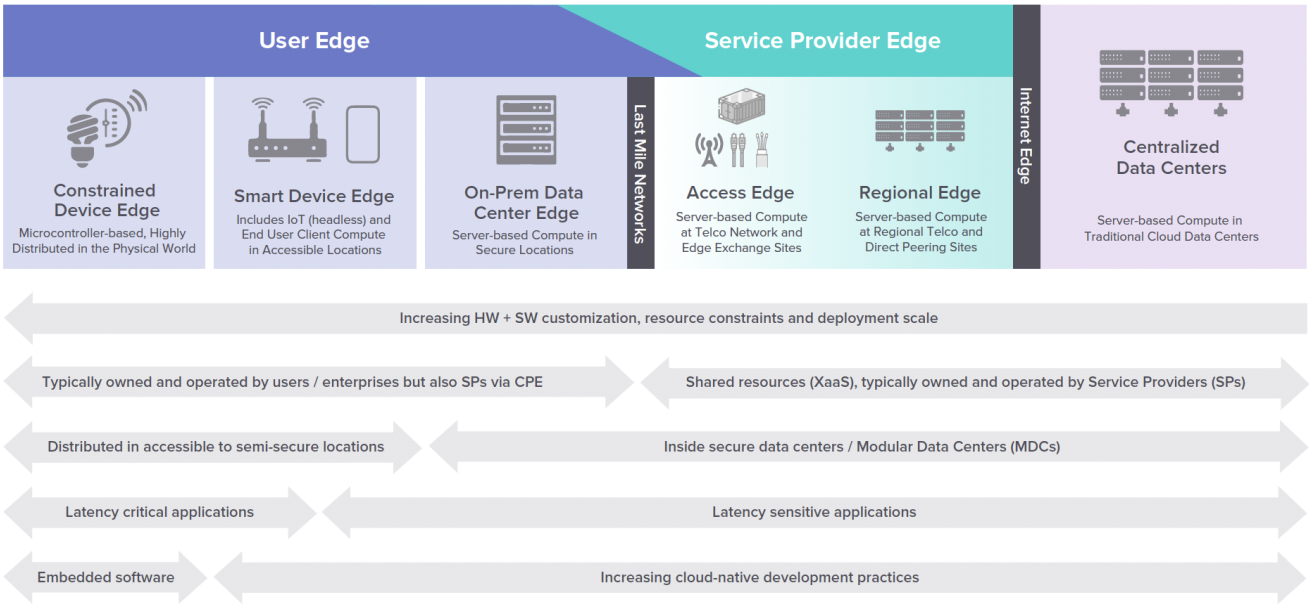
Kubermatic measures its software in a similar fashion. How good are we at designing autonomous software systems to do the tedious work for us, when it comes to integrating and managing infrastructure as well as software alike? Are we able to put the cherry on top and move software workloads without restrictions, but with highest security standards? Are we able to leverage the power of automation to our high standards? The target picture is quite complex:



What is it about Edge that makes it challenging?

Edge computing is unlocking many new business opportunities in areas like manufacturing, retail, healthcare, and telecommunications. Edge brings the service and the user closer together, in order to enable completely new services or enhance the user experience of existing ones

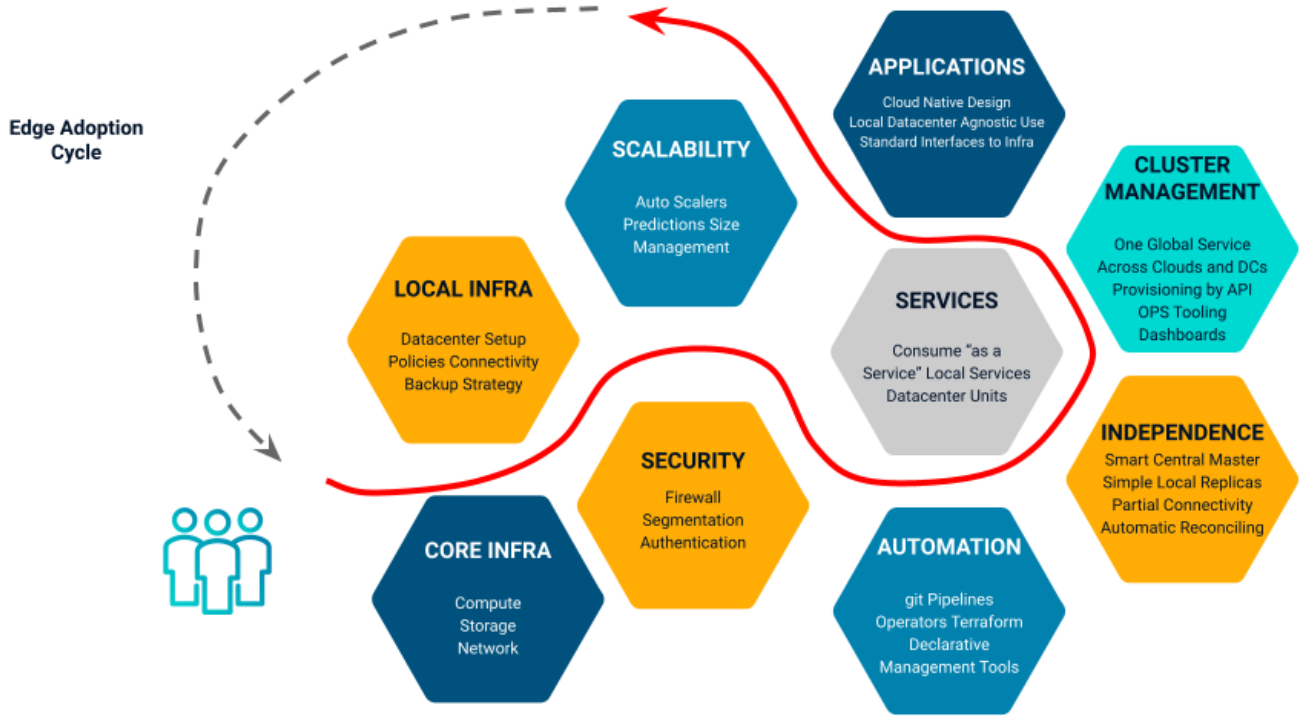
Most examples are network sensitive applications like live interactions, augmented reality, connected car, autonomous driving or manufacturing. The Linux Edge Foundation created a good image:



However, managing the scale, distribution, and life cycle of applications at a large number of edge locations requires fulfilling the following criteria:

1. **Security**: It is critical to ensure effective isolation of workloads to ensure non-interference of resources and data in a multi-tenant setup.
2. **Resource Management**: Operators must be able to manage centrally. Further, in case of a data breach, the system needs to be able to operate remotely and independently.
3. **Operations**: An effective edge system requires a high degree of automation and self-healing capabilities to reduce the impact of low network bandwidth, latency or outage.
4. **Open Standards**: Edge systems are characterized by their ability to deploy new workloads rapidly and integrate them with a variety of different environments.

5. **Stability & Predictability**: Edge compute platforms should behave predictably in different scenarios to provide a consistent and stable user experience.
6. **Abstraction**: Edge systems must offer an abstraction layer for infrastructure components to support portability of applications and workloads across platforms.



Summarizing, manufacturers need to consider the following topics as the most important part of the picture:

Key Requirements for Manufacturing Edge

- Hosting of Manufacturing Software
- Run everything inside an isolated manufacturing network
- High Grade of Automation
- **Independent** on operation of your microservice
- **High Availability** on distributed locations
- End-2-End Operation **across all Levels and Departments**

Open Source to the Rescue

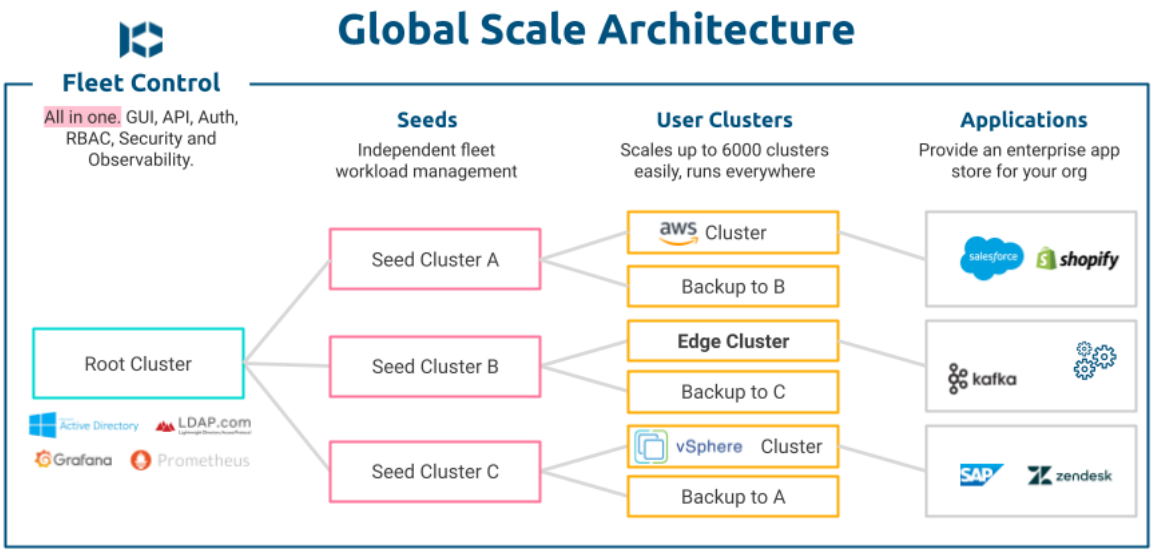
In order to satisfy these needs, proprietary software is often disqualified. The vendor ecosystem is never broad enough to cover all (edge) cases and even SAP centric customers invest heavily into customizing the vendor software - it is an integral part of the success of SAP to be able to ABAP (Advanced Business Application Programming) everything to your needs.

In 2022 it is a rather safe bet, that the open source community has touched a problem your organization faces and even delivered an already working solution, or it is likely that you can find a like minded team to overcome an obstacle, too large to overcome on your own. The customization and flexibility to adapt to new challenges brought us the two famous quotes “software is eating the world” and “open source is eating software.”

Navigating Edge with Kubermatic Kubernetes Platform

Kubermatic Kubernetes Platform (KKP) is an open core software available as a non-paid community and paid enterprise version. KKP solves the following problems:

- **Ease of Use**
  - in order to reduce cognitive load for developers and operators for running hyper automated systems at scale
  - Change whenever change is needed, no detriments when adopting
- **Deployment Flexibility**
  - Declarative approach to managing software and hardware alike
  - Gitops and API First a key
- **Availability on any infrastructure**
  - Cloud, Datacenter and Edge
- **Unparalleled density and resilience**
  - Management is independent from workloads
  - Resilience is built in by design with multiple seed clusters
- **Security and Observability**
  - Are integrated in one flexible open source solution



Manufacturing Edge

KKP for Manufacturing Edge Computing helps teams tackle these challenges by automating IT operations from the infrastructure to the application. KKP scales dynamically with your needs so you can start small and grow as needed without any downtime or data loss. You might use it to set up a single node that houses an application then scale out when demand increases. Or you could use it to run many different applications on one physical server simultaneously without having them interfere with each other's resources. And here is how that looks like in an abstracted form, but with network isolation and resource handling:

How Does It Differ From Other Automation Tools?

In many aspects, other popular open source tooling allows to achieve similar results, but these outcomes are not equal.

The biggest difference is that Kubernetes is both software and infrastructure. Kubermatic Kubernetes Platform now takes these principles to an extreme and allows to do all that at scale with security, observability and ease of use combined on both layers.

One Tool Only

Terraform is great at infrastructure but **cumbersome at software**, you patch the hole by adding a variety of tools and connections that **need to be adjusted manually.**

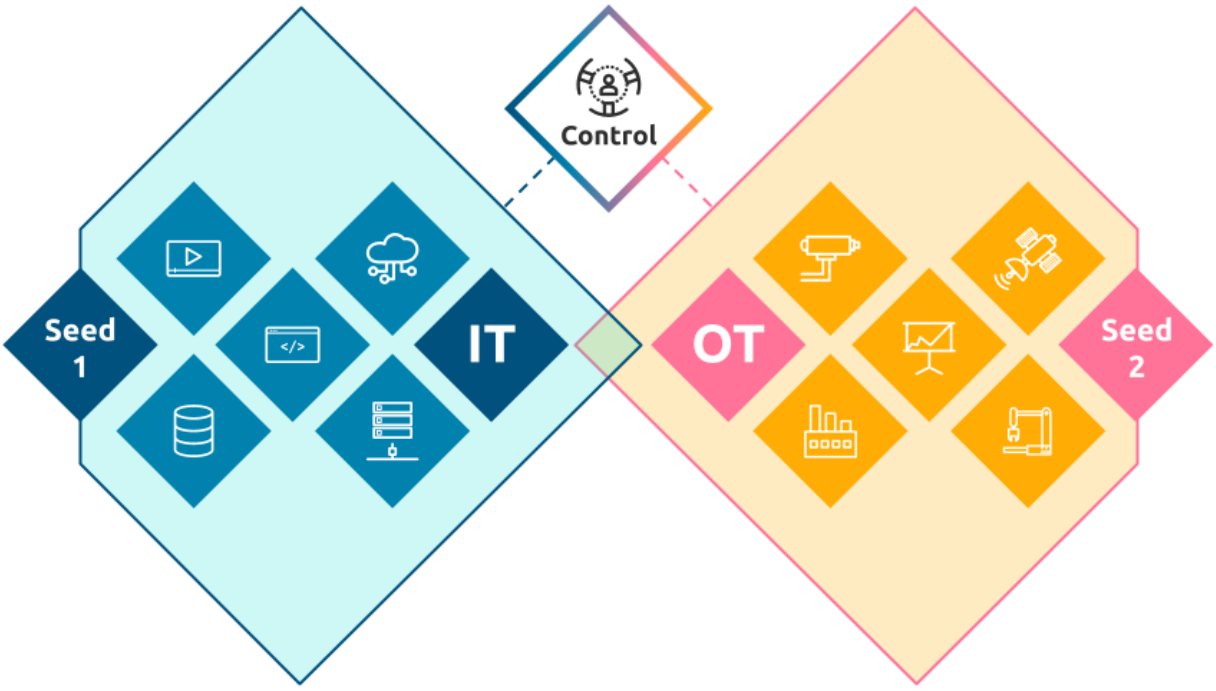
Kubermatic **is** both **infrastructure and software**. Zero Downtime updates for a fleet on both areas with one command!

Now that the context, requirements and architecture are layed out. The only task left is to display an actual deployment configuration that delivers everything at once.

Bridging IT and OT

Kubermatic has a unique capacity to work in IT and OT. Especially the control plane that can work in air-gapped (environments without internet access) setups are beneficial to OT environments. Whereas in IT connectivity is always important, in OT security concerns prevail. IT and OT can run different management clusters (seed clusters) and can configure these zones independently.

The building blocks of running a shop floor like traceability, analytics, process automation and many more can be put in containers or virtual machines. The embedded solutions are then connected within the user edge interface via standard network or specialized production networks. AMD64 and ARM workloads are on the OnPrem Edge onwards up onto the cloud. The highly specialized workload that resides in embedded systems or large scale production machines should remain inside of these and will be connected as demanded. The core requirements to separate workloads not only between IT and OT, but also within different OT zones are easy to manage. Network configurations can change dynamically or enforce pre configurations.

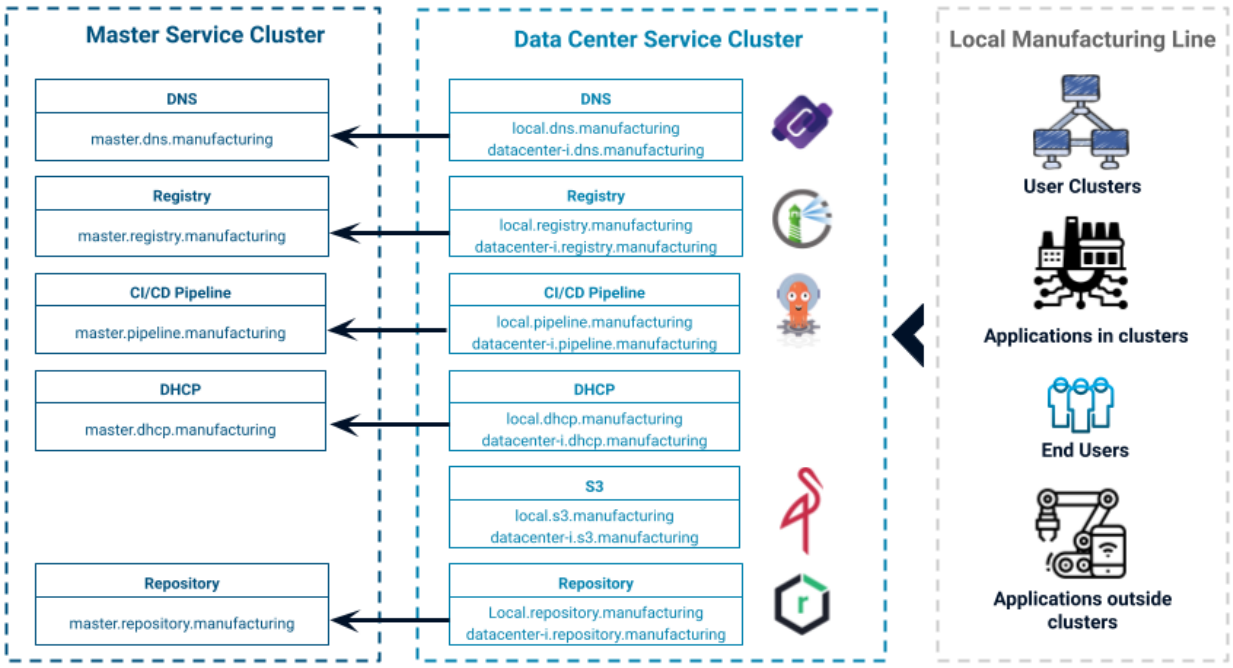


Real World Scenario

Achieving ease of use, security, observability, efficient operations and adaptability is only viable when delivering a distinct set of services out of the box. Kubermatic designed a service cluster architecture that runs alongside its multi layer management architecture in order to do just that.

The Service clusters deliver needed shared services like DNS, CI/CD, Container Registry, Repository, DHCP and S3 Storage. These service clusters can be configured to run locally without connectivity to the outside world (air-gapped scenario) and can be updated periodically with minimal risk of compromising the most sensitive infrastructure and data.

The key to this approach is to take air-gapped as the default case, not as the exception.



Take a deep dive into the depths of manufacturing edge in air-gapped and hybrid cloud scenarios with Kubermatic anytime.

Our experts will consult you, alongside our product portfolio, to help you become the manufacturing edge leader in your industry.

Global Leaders Work With Kubermatic

Get started with Kubermatic Kubernetes Platform

Contact our sales team at sales@kubermatic.com or visit our website at kubermatic.com