www.pipelinepub.com
Volume 16, Issue 11

The Road to Open Edge Computing

By: Ildiko Vancsa

Humanity has depended on tools from its earliest days. As we've evolved, tools have turned into what we call technology, which has had its own trajectory of evolution. This is true for cloud computing, as edge computing is the next step forward, bringing the cloud closer to people's everyday lives. This progress requires evolution in the assembly lines that support it.

Computational power surrounds us. It's on the fields that grow the vegetables we eat and in the factories that package food and make the plates and silverware with which we eat it. While our use of computers proliferates and expands, the evolution to enable further functionality never stops. Neither does progress toward connecting our ever-growing number of endpoints, simultaneously.

Cloud computing brought us the possibility to utilize resources in large data centers in a flexible and agile way. This is in stark contrast to workloads that ran on dedicated hardware, which provided an environment more suitable for applications but was wasteful and rigid. Even industry segments like telecommunications chose to experiment with cloud computing, and many operators and vendors went even further— not just deploying a cloud solution but choosing a platform with mainly open source components in it.

While there are cases in which cloud software on the back end is a natural fit, it rather comes as a surprise when your service provider (SP) picks this direction due to principles such as high availability, or five nines. This represents the service level agreements (SLAs) on the availability the SPs are required to provide for their services. Five nines translates to five minutes of downtime per year, and while it is very annoying when your favorite webshop is not available, it won't threaten your life like not being able to make an emergency call because the network is down.

Not for reproduction or distribution. © Pipeline Publishing, L.L.C. All Rights Reserved.

And yet the telecom operators are now paving the road for the next generation of cloud computing that many people just simply call edge. Let’s take a closer look.

The edge scene

Edge computing has become a familiar expression, but there isn’t consensus on what the “edge” in it refers to, exactly. You’ll receive as many answers as people you ask. The only commonality of the different edges is that they are all on the edge of something.

Edge computing depends on the paradigm of distributed systems and amplifies the scale to a level that wasn’t available before. The aim is to take both the computational power as well as the flexibility available in large cloud data centers closer to the users, be they humans or machines or the combination of the two.

This leads us to the far edge, the device edge, the large edge site, the small edge data center, the aggregated edge, the access edge, and so forth. These are all valid terms and make sense once you put them into context.

If you aim to give one definition to what edge itself is, you would need to bake the context, the use case and some of its characteristics into it to make sure those who use it will have the proper view and understanding. All these details make it impossible to provide one description that holds up in all circumstances, which makes the aim and process to create one definition unnecessary. At the end of the day it is *your* edge. See Figure 1.

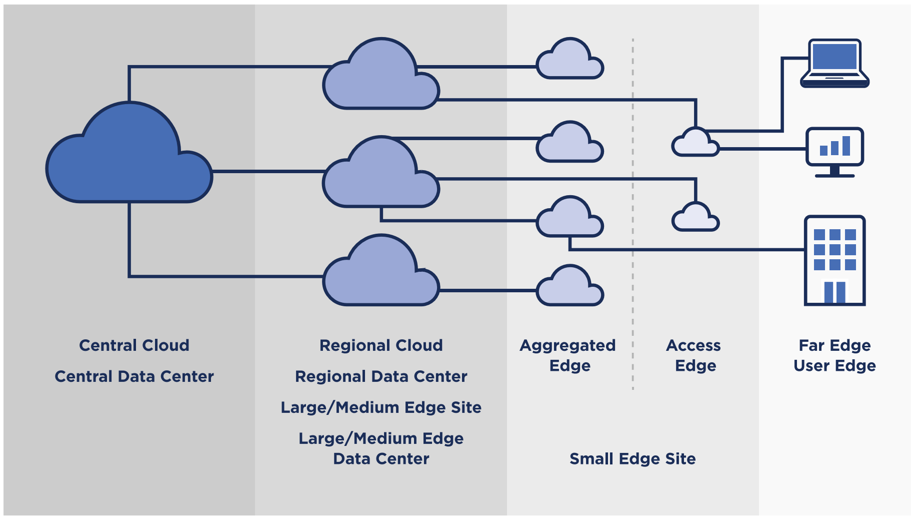


Figure 1: Definitions of “the edge,” in context
(*click to enlarge*)

One thing is common in all edges, which is that they are all part of a distributed system, the scale of which is greater than ever. This puts new requirements on your shoulders to solve and highlights imperfections in existing solutions.

Architecture options and where to find them

Not for reproduction or distribution. © Pipeline Publishing, L.L.C. All Rights Reserved.

The evolution of cloud computing doesn't take cloud out of the picture; it only restructures it, on certain occasions. In most cases, you will find that edge computing extends the already available cloud data center to bring more functionality to the end users. The environmental circumstances, unfortunately, will not change by this new method of organizing the available resources. You will still need to fight unreliable networks and will have workloads running in rural areas, which are sometimes hard and always expensive to access, but we still need to minimize latency, maximize bandwidth and take care of the lifecycle management of both infrastructure and applications.

The questions come naturally: How does one overcome all these challenges? And Is there an ultimate solution out there?

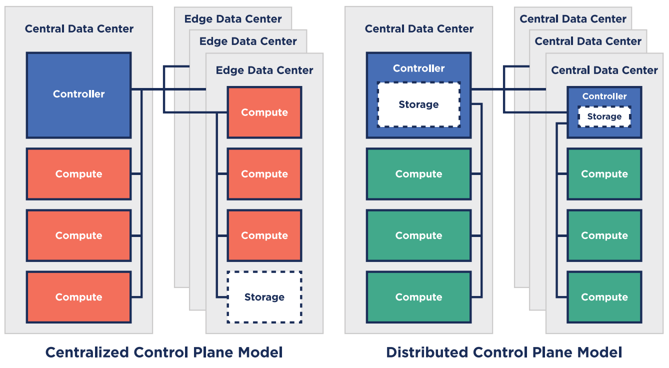
No matter how hard you try to find it, there is no one-size-fits-all solution that would satisfy every use case. The simple reason why comes from the aforementioned story about your edge not being unique but just different enough to make next steps a little more complicated than it first seems.

You can probably find components for your edge use case on the shelf of every vendor by now, whether they sell software or hardware packages or both. The new terms brought new business to the table along with the next questions: Which pieces do we choose and how do we fit them all together?

While edge computing is in its early stages, it already has a significant footprint in the open source ecosystem. Collaboration is crucial in this area because of the variety of building blocks that can be combined in numerous ways to fit an endless number of scenarios. Your AI and machine learning application can control robots on a factory floor or the life cycle of shrimp on a farm while being connected to a cloud infrastructure either locally or remotely.

And when we talk about connection, you immediately realize that the forever unsolved challenge of interoperability is under the magnifying glass here. Along with standardization, open source software development methods are most efficient in overcoming this obstacle by having APIs and interfaces openly defined and available to examine and test. The various groups can also work together to integrate many pieces of the infrastructure before any of it would hit your labs for a trial.

Groups such as the OSF Edge Computing Group are looking into use cases and requirements to find commonalities in the requirements and build architecture models that help development communities to evolve and test the pieces you will need to build your infrastructure on the available footprint you have.



Not for reproduction or distribution. © Pipeline Publishing, L.L.C. All Rights Reserved.

Figure 2: Model comparison
(click to enlarge)

Current architecture models all build on cloud data centers which are connected to edge sites of different sizes and varying capabilities. This often results in a spider-web-like graph. The scale of these webs can grow extremely large, especially in telecom use cases where you need to balance out how much functionality you put on the different dots and what it is going to cost you when you need to manage and maintain it. These observations have recently led to common error cases that may have specific behavioral needs that the infrastructure has to provide, which also give you hints about certain decisions. The most common scenario is losing connection between an edge data center and the central cloud. Depending on your use case, the autonomy you need in your edge sites differs.

Having the workloads still running and available on the disconnected site is the easier challenge to overcome, but if you need to be able to start new instances or provide user management functionality, you will need control functions available locally. The two main options to choose from are the Centralized Control Plane model and the Distributed Control Plane model (see Figure 2, on the previous page).

As you can see, the footprint of the two options are quite different when it comes to the edge data centers. It is also more complicated to manage and orchestrate all the control functions that provide the autonomy on the edge as you may need to synchronize user information and other bits of metadata throughout the infrastructure to maintain a consistent view.

You can find several examples that realize the above models. For example, the Distributed Compute Node (DCN) option of the Openstack TripleO project brings you the centralized architecture, or StarlingX is applicable if you are in need of a distributed setup.

A practical example

Let's take a closer look at StarlingX, which is an open source project supported by the OpenStack Foundation.

This project is a great example of the integration work that is needed to provide a flexible and robust edge platform. It also uses building blocks that you are most probably already familiar with, such as Linux, OpenStack, Kubernetes, Ceph and so forth. This points towards the aforementioned evolution path by providing you the pieces of a traditional cloud computing platform while giving you the option to deploy selected services on edge sites to get the required functionality (see Figure 3, below).

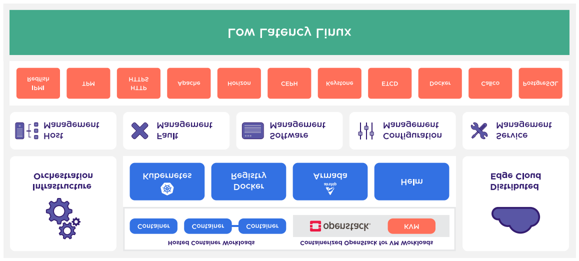


Figure 3: StarlingX
(Click to enlarge)

© Pipeline Publishing, L.L.C. All Rights Reserved.

Not for reproduction or distribution. © Pipeline Publishing, L.L.C. All Rights Reserved.

The project is utilizing containerization both for the platform services (to get flexibility and easier management) as well as for workloads, where it is applicable. As you can see on the diagram above there are a couple of services in the architecture that are responsible to manage the lifecycle of the hardware as well as software infrastructure, including the Distributed Edge Cloud component that is responsible for keeping your edge sites in sync.

The community has been focusing on requirements such as small footprint and high performance, which are both crucial for most edge use cases. However, evaluating success always depends on the use case and its specific demands.

Taking it to the edge

Edge computing is breaking down barriers between industries by taking computing power to cars, factories, fields and your home. The lines disappear between solutions as well, and new goals for operators require flexibility, agility and the ability to integrate as the environments are growing organically to take the edge always a little further out.

As the business models evolve, the cost implications will also become clearer, which will further encourage the industry players to participate in the open source efforts and work on the building blocks together.

© Pipeline Publishing, L.L.C. All Rights Reserved.

Not for reproduction or distribution. © Pipeline Publishing, L.L.C. All Rights Reserved.