Running an Overlay Network in a Multi-tenant Setup —Challenges of IIJ GIO Infrastructure P2 Gen.2

2.1 P2 Gen.2—IIJ's new-generation IaaS

On October 1, 2021, IIJ launched IIJ GIO Infrastructure P2 Gen.2 (P2 Gen.2 for short), a new IaaS that fully combines and takes advantage of the characteristics of the public IaaS and private IaaS developed and provided under the IIJ GIO brand. P2 Gen.2 is a new-generation IaaS designed to confer the full benefits of public clouds while also taking advantage of VMware vSphere-based private clouds.

To ensure customers can continue using P2 Gen.2 into the future, we adopted an architecture geared toward the next generation of services. This report looks at the development of the P2 Gen.2 service and the ideals and reality of its operation in the field with a focus on the network.

Our aim in developing and running P2 Gen.2 is to provide a service platform that will continue to be useful a decade or more from now. Today's rapidly changing world makes it impossible to predict what will happen 10 years ahead. Yet we believe that services should remain available for as long as customers want to use them. The development of new systems based on cloud technology under the banner of DX (digital transformation) is on the rise. But maintaining and modifying existing systems to support DX is also a key mission for enterprise IT teams. We developed P2 Gen.2 to address this challenge by providing a third option for cloud computing that can not only accommodate on-premise systems and private clouds that remain in operation but also make it easy to migrate from public clouds as well as.

P2 Gen.2 is an IaaS offering that allows customers to use resources in the form of flexible server resources. The resource pool is virtualized, customers focus on system operations within the resource pool, and IIJ runs all of the functions in the layers below it. With IaaS run in individual VM instances, the size of the VM is made to conform to the instance model specified by the cloud service provider, but with P2 Gen.2, customers are free to create VMs in the resource pool. This means that customers can transfer

the specification of the machines running in their own environments as is. Naturally, this design makes it possible to bring in images or migrate VMs from the existing environment as is via V2V and P2V migrations.

P2 Gen.2 provides an image provision mechanism for deploying VMs; backup, network, and migration functions; and file servers for storing files. The hypervisor and hardware such as servers, storage, and networks are abstracted in a form that customers are able to visualize, so they do not need to worry about these elements. This eliminates the workload encountered when renewing the hypervisor and hardware that results from handing ESXi directly over to the customer, something that had been a major issue with the Virtualization Platform VW Series, a vSphere-based service provided on IIJ GIO.

We adopted VMware Cloud Director (vCD), a VMware product for service providers, and hid the hypervisor (vSphere) layer from users. This made it possible to offer resource control permissions that are as flexible as with vSphere while allowing IIJ, as service provider, to take on the hypervisor and hardware lifecycle management role. Defining a new joint-responsibility model like this made it possible for IIJ to manage and operate the hypervisor network. We adopted VMware NSX-T Data Center (NSX-T), which enables integration with vSphere, to run the hypervisor layer network efficiently, and this significantly improved the IaaS network. With P2 Gen.2, in Layer 3 an NSX-T overlay network sits on top of an IP fabric underlay network, and each tenant's network is completely separated from the others and provided as a VPC (Virtual Private Cloud). Combining this with the operational knowledge in large-scale server pools we amassed through IIJ GIO, we have made it possible to allocate resources to users in a manner independent of the allocation of physical computing resources (CPU, memory, storage) (Figure 1).

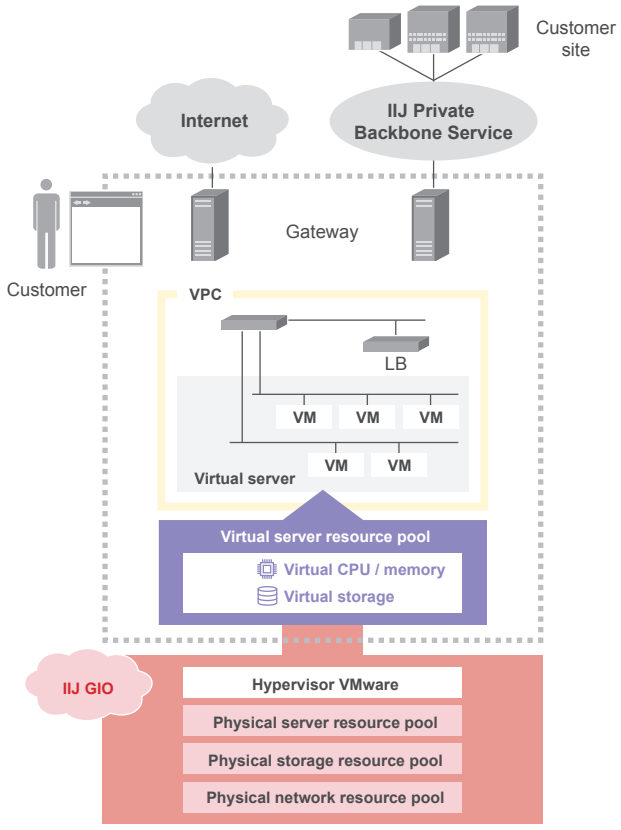

Figure 1: Flexible Server Resources Overview

2.2 Overlay Networks Using SDN Technology

Overlay networks are nothing new. Along with VLAN (Virtual LAN), they are a long-standing network virtualization method. In the past, they were often used for point-to-point connections due to protocol specifications/constraints and for operational considerations, but with the evolution of SDN (software-defined network) technology, they can now be used in large-scale networks. Encapsulation protocols like L2TPv3, VXLAN, and Geneve are used to build virtual L2 networks. SDN technology uses software to control the networks, and is made up of the control plane, which controls what happens between devices, and the data plane, which forwards the actual frames and packets. Using software to control what happens between devices makes it possible to manage the configuration centrally. This creates an environment in which administrators can automatically configure devices according to how the network is defined, instead of having to change the configuration of each individual device, so configurations can be changed quickly to meet the requirements of large-scale networks. This makes it possible to have more than 4,096 network segments, the limit imposed by IEEE 802.1Q VLAN, something that has posed issues in large-scale networks that use many segments. It also enables the efficient control of north-south traffic (traffic between devices and the network center) and east-west traffic (traffic among devices) within virtualized networks (Figure 2).

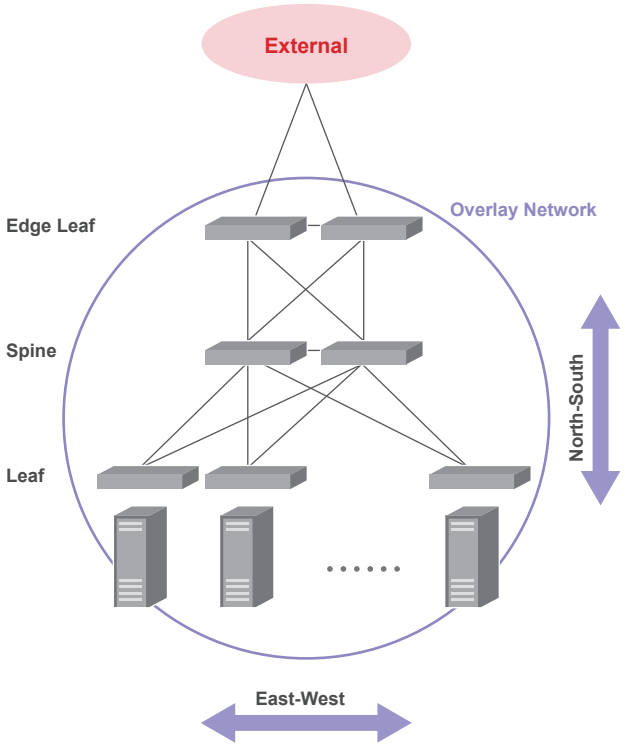

Figure 2: Overlay Network in a Large-scale Environment

2.3 Benefits of Overlay Networks with VMware NSX-T

Along with server virtualization technology, network virtualization with overlay networks is one of the core technologies underpinning the P2 Gen.2 infrastructure. Service networks can be configured separately from the underlay network that lives on the hardware layer. Using NSX-T for network virtualization on the IaaS layer of P2 Gen.2 provides a flexible network configuration and loose coupling with the physical layer. When setting up networks with a view to expansion from minimum scale up to hyperscale, they tend to become complicated in terms of physical device specs because design and sizing need to be chosen in anticipation of that expansion. But being able to control expansion via overlays confers considerable benefits on the operational front.

An IaaS also needs to handle multi-site configurations across multiple data centers. Using infrastructure across multiple, physically separated sites facilitates disaster preparedness and makes it possible to obtain the necessary computing resources without needing a large-scale facility. Virtualizing the network between sites and making it an overlay network enables you to configure networks independent of their location. Physically separated environments involve more considerations to take into account than with single-site configurations, such as network latency and bandwidth between sites, and this increases design and operational difficulty, but ensuring proper resource monitoring and control of systems based on the monitoring data makes it possible to set up flexible networks.

Network virtualization remains uncommon despite these benefits. With traditional networks where the configuration was expected to remain static, administrators were able to continue maintaining and running systems without the benefits of network virtualization. In recent years, however, there is a need for IT infrastructure as a whole to adapt flexibly and swiftly to rapidly changing and unpredictable requirements. Network virtualization will likely become more and more important in this landscape.

IaaS solutions need to be multi-tenant (accommodate multiple customers). Centralized control and automation via software is also needed in order to provide functions separately to the tenants and to maintain security and the specified service level. Network virtualization makes it easier to link together the other components of the IaaS solution as well as monitoring systems, facilitating proper control of a multi-tenant environment in which each tenant operates their system in accord with their system requirements. NSX-T also implements VRF (virtual routing and forwarding), and VRF can be used to enhance equipment efficiency. In multi-tenant environments, the need to divide up routing domains among the tenants posed a resource constraint. VRF makes it possible to increase infrastructure consolidation.

Interface control is centralized into NSX Manager, NSX-T's management component, making API-based operations and automation easy. This also helps to reduce development hours spent on making the service's backend applications work together.

When an overlay network is set up as an IaaS solution, network exits are needed. NSX-T also cannot communicate outside the NSX-T domain (under its control) with the NSX-T encapsulation protocol Geneve, so to communicate with the external world, you need to convert it to a VLAN. We had not fully automated external connections with IIJ GIO, and a major issue was that users were unable to change network configuration in a complete form when migrating systems. P2 Gen.2 improves this situation by automating route configuration for connections with the IIJ Private Backbone Service. NSX-T can also be run in a bare metal configuration with NSX Edge installed directly on a server with no virtualization, and while this is fine in performance terms, it is not really up to the task of multi-tenant configurations. Many challenges also present themselves particularly with external connections, such as downtime during software upgrades. So many necessary areas of improvement remain, and we continue to work on resolving these issues.

2.4 Operational Issues and Solutions

Ahead of the October 1, 2021 release, in 2020 we conducted a small-scale rollout of the NSX-T SDN-based overlay network, along with NSX-T's NFV (network function virtualization) capabilities, for some data center functions. This revealed a whole host of issues, which we continue working to address as part of our operations. Below, we go over the issues we discovered and their solutions.

Creating an overlay network using SDN requires a more complicated configuration than with conventional networks. The configuration has two layers: the physical layer (underlay) and the IaaS layer (overlay, logical layer).

Another point is that to get the most out of the NSX-T architecture, you need to be well-versed in vSphere operations, so in addition to networking, the engineers also need to be able to work with server, hypervisor, and virtualization technologies. This is a departure from the past practice of specializing in specific areas—à la conventional network engineers and server engineers—and instead, the engineers need to be familiar with the entire technology stack used in building the infrastructure. When we started out, our team was small but included people with a strong knowledge of VMware technologies and people with experience running large-scale service infrastructure, so as a team, we had the capabilities to handle ongoing development and operations. The most important part of getting engineers with differing areas of expertise to work both autonomously and in collaboration with each other as a team is to have a vision for your products and services and strive continuously to make them better based on the DevOps philosophy.

With P2 Gen.2, we revisited infrastructure control from scratch to ensure it would work the way we needed it to. We put as much effort as possible into streamlining operations and automation. Under DevOps principles, we built a Git-based CI/CD pipeline, and the engineers control the majority of elements that make up the IaaS,

not just the VMs, under a policy of making changes to the operations and configuration of the infrastructure without directly manipulating the environment (Figure 3).

While there is a lot of knowledge and software out there for controlling VMware products, the de facto standard in server virtualization, it is not quite adequate for total control of multi-tenant infrastructure in the form of a SDDC (Software-Defined Datacenter), so some of the tools and libraries we use were developed from scratch.

We are not only simply carrying out operations tasks through these efforts. We have also laid the groundwork for implementing SRE (site reliability engineering).

Taking on board agile values and methods like this can facilitate more flexible and agile infrastructure operations, which can otherwise tend to be fixed and inflexible, but you also need to be keenly aware of the opposing relationship between agile principles and facility operations. Procuring physical equipment involves delivery lead times, and setting up physical equipment in the data center is not something that you can do via software control alone. Configuring facilities involves some level of physical human interaction, so making sure the associated lead times do not become a bottleneck is key. Having an abundant inventory of resources can solve this issue, but this then leads to inefficiencies because the overall resource utilization rate falls. Also, out-of-resources conditions are unacceptable in cloud services. So resource management, monitoring, and demand forecasting are crucial. Monitoring and demand forecasting should be done not just on the overlay, it should also aggregate and facilitate visualization of the various underlay data sources, including servers, network equipment, and storage. Since the October 1, 2021 release, we have been collecting and analyzing data on actual usage conditions to enable just-in-time resource inventory deployments based on demand forecasting.

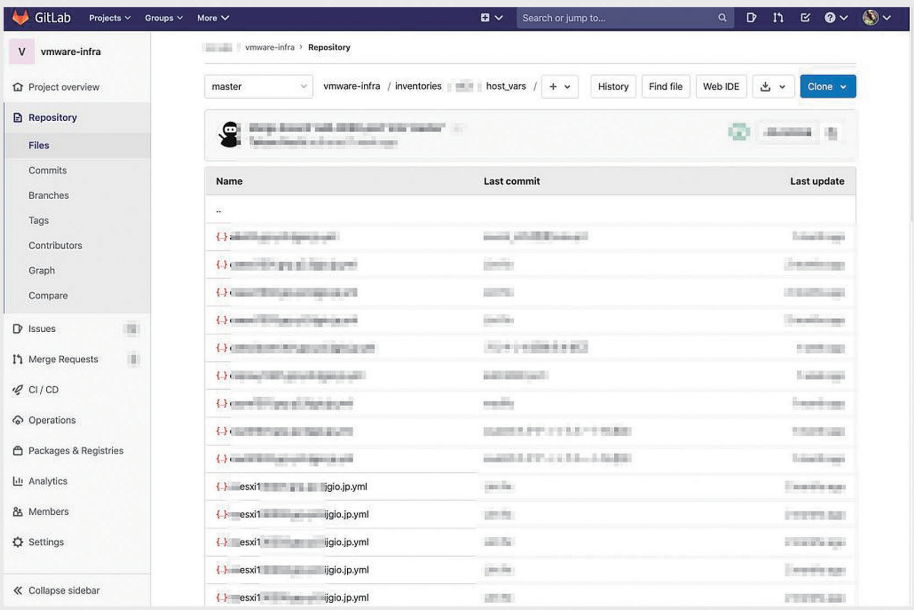

Figure 3: The IaaS Components are Managed via Git

2.5 Looking Ahead

P2 Gen.2 still only has the minimum functionality, and we plan to continue developing it to serve customer needs. The use of computing resources in the form of cloud-native IT systems has become commonplace, and a more agile digital infrastructure is now being demanded. Our first priority is to implement the features needed to ensure that customers will continue to use the service, but we also want to be able to address future needs beyond that.

In addition to migrations into the cloud, the need for edge computing and distributed computing that deploys computing resources close to where users are is also rising. IIJ's role is to connect the cloud at the core, the distributed edge nodes, and the network in between. These interconnected elements will allow IIJ to deliver high-quality services on edge nodes closer to where the customer is actually located. In addition to conventional networks, 5G mobile networks and the like can also be used to connect the resources. Linking the core and the edges requires not only VMware-based workflow management but also application container-based resource management, so also providing resources in the form of Containers as a Service (CaaS), allowing applications to be run inside containers, might be the optimal solution for customers (Figure 4).

Software control that brings the network closer to the status of an application is needed here. This goes beyond just resource services that provide VMs, and I think it can be achieved by evolving the network from the current

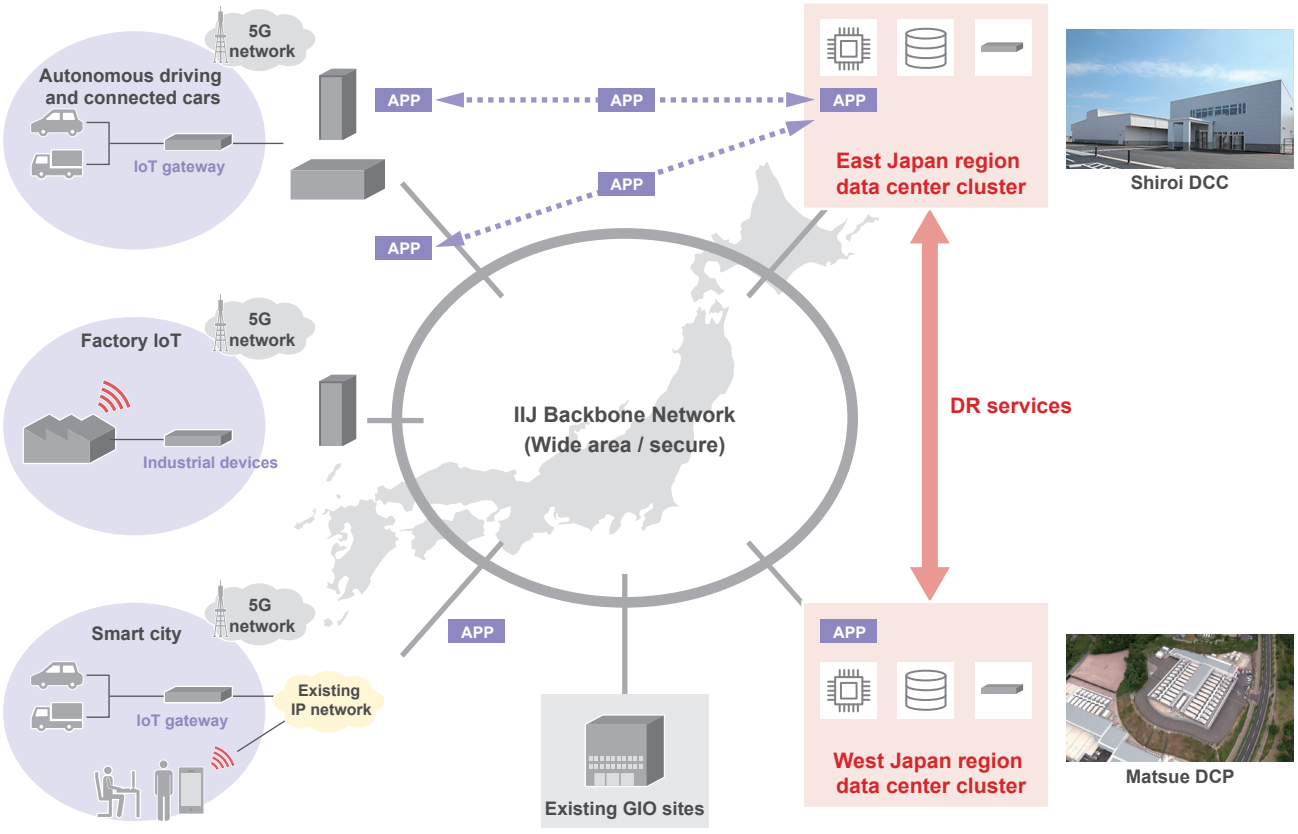

Figure 4: Looking Ahead

architecture into something that is even less dependent on physical resources. This will be difficult to accomplish simply by deploying resources in a compact form that can also be deployed on the edges. You also need to be able to configure the network flexibly so that it works together with the other resources.

Personally, I am interested in SmartNICs as a technology suitable for this sort of edge computing and distributed computing architecture. VMware is working on Project Monterey, which runs ESXi on Arm-processor based SmartNICs. The hope is that managing SmartNICs as NSX-T domains will make it possible to control and deploy networks in a manner that makes them equivalent to applications.

To make this sort of world a reality, we are also working on a project to use the technology stack we have developed not just for data center purposes but also as an edge computing platform. We have set up a micro data center outdoors at IIJ's Shiroi Data Center Campus, and we are conducting tests in the area of cloud and edge computing as well as multi-access edge computing (MEC), with our objective being to develop the technology further along with local 5G, IoT, and other use cases. My hope is that evolving network services in this way will make it possible to provide customers with unique infrastructure of the sort that only IIJ can deliver.

Takehiro Yamamoto
Infrastructure Service Design Manager, Cloud Services Division 1, IIJ System Cloud Division.
Mr. Yamamoto is engaged in the planning, development, and operation of cloud-related services.