Constellations Podcast

Episode 86 – Space Innovation, Dynamic Ground and OpenSpace™

Speaker: Greg Quiggle, VP, Product Management, Kratos – 33 minutes

John Gilroy: Welcome to Constellations, the podcast from Kratos. My name is John Gilroy, and I'll be your moderator. Our guest today is Greg Quiggle, Vice President of Product Management at Kratos. Now, there has been a lot of exciting innovations in space from multi-orbit satellites, mega-constellations to software-defined payloads. Today we will discuss the technology revolution that is occurring on the ground, that is critical to enabling these developments in space. Northern Sky Research, a global leader in satellite and space market research explains the satellite ground network infrastructure is on the cusp of a fundamental change. Ground network technologies must shift from purpose-built proprietary hardware architectures to software-defined, flexible, and extensible virtual platforms.

Addressing this challenge, Kratos has developed the first fully digital virtualize software-defined and orchestrated platform in the satellite industry. It's called OpenSpace. OpenSpace is a centralized platform that uses software control and programmability to automate functions across the ground system from transport, infrastructure, and management capabilities to improve scalability, resilience, and lower costs. With us today, Greg Quiggle, he is focused on enabling more dynamic ground system operations by leveraging the power of software-defined networking technology. Greg has spent over 25 years conceptualizing and executing successful corporate-level product and technology strategies within the communication industry. Greg, what is changing in the satellite industry that's driving this revolution on the ground?

Greg Quiggle: Hey John. I would tell you that most of the ground innovation today is really being pushed by an incredible amount of innovation in space. Some would say we're actually in an era of innovation in the space industry, and you see that in the form of multi-constellation satellites. It used to be primarily GEO. Now you see mega-constellations at low earth-orbit. You see middle earth-orbit satellites. You see the move to proliferated LEO, you see also software-defined payloads, payloads that no longer launched in a fixed state for a period of 10 to 15 years. They actually reconfigure themselves based on changes in user demand. All of these things are really leading to unprecedented gains in terms of bandwidth that's available over space satellites, flexibility, and also in terms of performance. These are having a notable impact then on the way basic ground systems really need to operate.

John Gilroy: Well, Greg, we're going to talk about movies now. We're going to talk about the movie, The Perfect Storm. It sure seems like with the proliferation of satellites out there, all these different types of systems, LEO, MEO, and GEO, it seems like

this is a perfect storm to change the way that we accept information, change the way we get these communications from the satellites, doesn't it?

Greg Quiggle: Yeah, absolutely. I would actually argue that if the ground systems don't change all of this great innovation in space will actually go wasted. You know, the ground systems simply don't have the scale or the flexibility or the performance necessary to be able to react to these changes in these constant new features that we see in the space domain.

John Gilroy: I get bombarded by all kinds of press releases, all kinds that are involved with technology in the cloud. Now a couple of weeks ago, there was Microsoft Ignite and I saw a press release about their ground station as a service offering. How does this as your orbital ground stations of service fit into this just crazy changing ground system environment?

Greg Quiggle: It's actually a very good example of a true digital dynamic ground platform that can react to these changes in space. Orbital is actually all about Earth observation networks and its initial manifestation. What it does is it allows the ground system to spin up, to take a pass from an Earth observation satellite, as it orbits over the ground station. The benefits to this really are significant to the Earth observation operator, but before you had something like Orbital in place, what would happen is the Earth observation company would have to build their own static ground system based on a hardware and because most EO satellites are LEO, the ground system would only be utilized at points in time when the LEO was actually in site, they move in orbit. What the Microsoft solution allows you to do is instead of dumping all of that capital into a dedicated hardware-based ground system, you literally can use as a shared entity, Microsoft antennas and infrastructure to take pass data as the LEO flies over the top.

Greg Quiggle: There's a large benefit right off the bat for the Earth observation companies and that they don't have to put a lot of CapEx into an asset that ultimately is underutilized. Equally as important, most of the imaging data that comes from an Earth observation satellite typically needs to be analyzed in the cloud. There's an added benefit herein that the data that's coming down from the satellite is already in the cloud through the Azure orbital network, as a result, it can be quickly processed with minimal added costs for ingress-egress data or latency.

John Gilroy: You used word capital and as you were speaking, I was writing notes about capital. I thought about the amount of capital that Microsoft's invested in this. It's just, it's billions of dollars they've invested in this technology. It's just amazing. When I think of Azure and Kratos, Kratos is a company that is helping to enable Azure Orbital. What role does Kratos play in terms of this more dynamic ground system?

Greg Quiggle: I would tell you there's a couple of key roles that we provide for Azure. The first is we actually provide some critical components for their ground stations. As the antenna is pointed to a satellite, the downlink is received, it's processed digitally so that it's able to be transported over an IP Ethernet network through into the Azure cloud. Kratos provides a lot of the infrastructure for that as a key component of what is our OpenSpace platform.

The other key thing that we do with Microsoft is we also have a marketplace presence in what that the marketplace presence does is these Earth observation providers that want to be able to process their downlinks. They can actually do transactions with Kratos, through Microsoft to set up all of the virtual network functions and service chains associated with processing the link all fully in the cloud.

John Gilroy: I've been doing this for 25 years you know. Just think about 25 years ago. Okay, in the future, there'll be a marketplace where you could get services for yourself. Would that even dream of something like that years ago. You never dreamed of that. Did you?

Greg Quiggle: No, and I would actually have to tell you, many don't think it's feasible today.

John Gilroy: Even today. Wow.

Greg Quiggle: You know what, you mentioned all of the infrastructure and investment that you see with these large cloud players and it dwarfs the satellite industry. A big part of OpenSpace is actually taking advantage of that huge amount of investment and technology and standards that live not only in the cloud industry but also in the telecommunications industry, the wireless industries. We want to embrace those standards and those technology investments, because in the end, if you do that, it actually allows the space industry to grow much, much faster.

John Gilroy: My wife's a Latin teacher, she talks about nomenclatura or nomenclature. Let's talk about nomenclature here, OpenSpace. So what is OpenSpace and what does it mean for our listeners?

Greg Quiggle: OpenSpace is a dynamic ground platform and really what makes it unique, there are a few things. The primary though, is it's really built upon the core fundamentals of network function virtualization NFV and software-defined networking SDN. There are four basic components to what makes up OpenSpace. As close to the antenna as possible for any of these systems. What we do is digitize all of the RF links and in EO case, it's a downlink and a TT&C or a Satcom network. It would be a two-way link, both uplinks, and downlinks. We digitize and package all of those RF samples into a standard called VITA 49. That

really allows us to move these digital samples really anywhere in the world, over a standard Ethernet or IP network. Once you get to a point where you want to process those samples, then we do that with virtualized network functions and virtualized network functions really are they're virtually equivalents of common components.

Greg Quiggle: You would see in a hardware in a teleport today, it could be splitters. It could be combiners. It could be modulators, demodulators, spectrum analyzers, matrix switches. All of these things today have physical components. What we've done with OpenSpace is we take those digital samples in over VITA 49, and we now process them as virtualized network functions. That allows us to really do this on standard x86 compute, which allows it to run a cloud native. These VNFs are spun up and spun down on demand based on changes in the network, changes in customer demand, changes in threat, and they're organized through something called a service chain, which is an SDN term. Service chains are essentially built by our OpenSpace controller, which is the third component of OpenSpace. It allows us to really apply business logic to these requests for services to dynamically instantiate service chains, which are really combinations of VNFs.

The last component is OpsCenter. OpsCenter is really just a unified management interface. It's a dashboard per se that really spans the physical side of the network, right? Some things physics get in the way you still need physical hardware, like the antennas and the amplifiers. It spans that physical domain with the virtual domain where all of the virtual network functions and service chains live. In doing that, you bring all of these things together. It really allows the ground system to be much more dynamic, which allows it to respond to all of these changes that we see in space. It also responds in similar fashion to highly dynamic networks that it interconnects with terrestrial and wireless networks.

John Gilroy: Well, Greg, one of the problems I have in life is I don't see the obvious. Let's look at the obvious here. The first word “Open”. We put the “open” in OpenSpace. What's the value of being open in this industry?

Greg Quiggle: Yes. If we go back a few years, I would tell you that historically the satellite industry has really been built on proprietary hardware centric networks. In the early days, that came from the state-of-the-art signal processing that was required to make these, these systems operate. In modern times though what's happened is the signal processing infrastructure has gotten so much better that the industry has really needed to move from these proprietary hardware oriented systems to things that more fundamentally leverage latest, greatest interfaces and techniques that are used in the trust real marketplace. Also, even recently with 5G with the wireless marketplace. What OpenSpace has done is it's taken this fundamentally different approach by leveraging a suite of these

interfaces that they've been heavily adopted and proven to scale in wireless networks and terrestrial networks. For example, all of our VNFs run in a standard container.

Greg Quiggle: They're all configurable via a standard called open API, which makes it very easy to configure and integrate with the VNFs. The VNFs all run cloud native within a variety of different resource environments. You do this through a standard Vim interface, virtualized infrastructure managers. For example, Azure has a Vim Amazon, AWS a Vim. VMware has a Vim, OpenStack has a Vim and they've all agreed to a standard such that virtual network functions that want to run in those environments they can leverage a common set of interfaces to do so. That's on the resource side. If you look on the service interface side, we follow a model called YANG and believe it or not, that stands for yet another next generation data model, right? Engineers if nothing else do have good humor. But ultimately that's similar in that it's a language or a data model that's been built that allows you to generically describe a service.

It distracts all of the product and technology details from the basic service request. I want to do a 10 megabit per second link between these two points. By leveraging these standard interfaces, like open API, Vims, YANG models and subscribing to standards bodies MEF with their lifecycle service orchestrator framework it not only allows our systems to scale much more quickly, but equally as important. It allows us to very quickly integrate and scale with these peer networks that live with terrestrial carriers like an AT&T or Verizon, but also with wireless operators as they move from things like 4G to 5G.

John Gilroy: Everyone's at competition, McDonald's, Burger King, Chipotle. So, so how is the OpenSpace platform different from other digitization and virtualization products in this specific world, in the satellite world?

Greg Quiggle: I would tell you right off the bat, it has to do with the fact that OpenSpace has approached this problem as a platform. As I mentioned early on, it's largely based on these fundamental principles of network function, virtualization and software-defined networking. There are many peers out there that have virtualized components, right? They might make a virtual modem, they might make a virtual spectrum analyzer. That in itself has been a step forward in our industry. What those things are missing though, is the ability to scale and be dynamic for these large networks because them and themselves are just one part of the overall product offering. What OpenSpace does, is it says, okay, even in a modest size network, there's going to be hundreds of thousands of these virtual products. Nobody's going to scale up. They have to manage each of them independently.

What OpenSpace does is it applies these fundamental principles from software defined networking and NFV to allow you to deploy these things dynamically

and react to very quick changes and the supply of the network more compute, less compute, or perhaps spectrum that's operating well versus spectrum that's not operating well. It can react to changes in user demand. Customers at any point in time could be drawing more throughput over a satellite versus less, and also it can react to threat conditions. Perhaps it's a military use case in theater where your adversary is jamming, a frequency band you're using, and you need the network to react to it. What OpenSpace does, is it takes advantage of this step of virtualization, but it really takes it to the next step by making all of those things work together in concert as a dynamic network.

John Gilroy: Greg thousands of people from all over the world have listened to this podcast. Go to Google and type in Constellations podcast to get to our show notes page here, you can get transcripts for all 86 interviews. Also, you can sign up for free email notifications for future podcasts. Now, Greg, a lot of people who are listening to this podcast may be technically oriented and we all know about x86 and containers, and maybe some of them know about Vim, V-I-M, but there are other people that listen to this podcast, maybe they are in higher levels of the organization and they have the proverbial Tom Cruise question. The Tom Cruise question is okay, Greg, you're technically competent, but show me the money. Show me the money. Can this help me save money? Is it going to reduce costs or both? Where does the whole financial aspect fit in here?

Greg Quiggle: I would tell you, it's a little bit of both this shift to dynamic software-defined networking most certainly translates to reduced OpEx, operating costs and CapEx capital over the life cycle of a given satellite fleet. It's been proven already multiple times in other industries, right? Cloud in general is a good thing for costs. Software-defined networking in general is a good thing for cost. This is furthered actually by the ability to align network spend with actual adoption of customers on a satellite. If you think of the way the industry works, an operator will launch a satellite. Typically, they have a 10 to 15 year life associated with that satellite. When the systems are largely hardware and static, they actually have to invest in the complete ground system, assuming 80 to a hundred percent adoption of the satellite day one.

So they have that large cost sitting there. It's not like they have 80% of their satellite filled with customers day one. Typically they have 25% of their satellite filled with advanced orders. That's wasted CapEx that you could be using on other things. When you move the network to software, what it allows you to do is apply cloud resources as you add customers. The amount of upfront spend needed is minimized. You put the minimum amount of spend in place to be able to light up the spectrum, but processing the spectrum can scale as you add customers, which really not only makes the overall life cycle cost less. It allows you to also just control the spend. As the revenue comes in, a portion of that goes to the cost of your network versus needing to spend all of that upfront.

Greg Quiggle: That being said, I think that's the cost side of the equation. There's also a revenue side of the equation in that with the open interfaces and scale that you get with OpenSpace. It does a couple things for you in terms of being able to very quickly reduce the time needed to design and deploy services with your existing customer base. If I look at a hardware-based system, if you want to deploy service to a new site, believe it or not, it could take a week or more in order for you to be able to do that. There's a lot of logistics involved in getting equipment in the right place, dealing with exports and imports, dealing with staff locally, to be able to do the job. I'm being kind when I say a week, it could easily be more. When the world went to virtualized products, that time went from a week, let's say a day or more.

That does take advantage of the benefits of virtualization. We've been experiencing that for many years with our quantum product line. What OpenSpace does is it takes it from a day or more to seconds. And the reason why that happens is because the network is actually orchestrating these things called service chains on demand. You're leveraging a common, generic hardware infrastructure, both in terms of compute and digitizers that can work really with virtually any antenna, any band, any amount of bandwidth. Through that, you can actually build a service chain in well, less than a minute.

John Gilroy: Wow. Now Greg, when you sit down with a prospective customers, they're going to of course ask about your technical background. We got those check boxes, then they're going to ask him the many money question, they're going to check boxes, and then they're going to turn to you and say, "Okay, who has used this besides me?" The early adopters are testing this platform. What kind of feedback have you gotten from them, Greg?

Greg Quiggle: Yeah. It's hard for me to give a lot of specifics given this forum, I'm obviously in public forum, you already mentioned the Microsoft announcement that we saw a few weeks ago. We're delighted to be a partner with Microsoft and already have pretty deep engagement with several customers. Beyond that I would tell you many of our quantum customers are actually already in discussions with us about upgrading and are at various stages in that process with OpenSpace. Again, quantum is primarily a set of virtualized products so as I mentioned, when they look at something like OpenSpace, it allows them to go from a deployment in a period of a day to deployment, literally in a period of seconds, as they add new customers to the system. It's hard for me to go much deeper than that, but I will tell you the initial response to our first use case, which is for EO sensing networks has been really, really good.

John Gilroy: Well, we're in the Washington DC area, recording this, sitting across from each other in the COVID crisis. We got to ask about the federal government here. What about this new approach to standards based and virtualized ground

systems? Does that match up with the needs of government agencies in the military?

Greg Quiggle: Absolutely. The government agencies actually are a really good representation of many of the challenges that you see with big terrestrial carriers. Very, very heavy user base, right? Very, very large networks and have to support a whole variety of use cases. For the same reason that these terrestrial carriers like the AT&Ts of the world went to software-defined networking 10 years ago, it's the same motivation that ultimately will drive our government customers and federal customers down the same path. Many years ago AT&T was also very much based, their network was based on very hardware centric systems. Lots of custom and proprietary interfaces that were vendor specific and kudos to them and others. They said enough is enough. In the wireless industry, you saw this move to a 3GPP as a way to drive standards within the wireless systems.

Before you got to a 3G, 2G systems were very much like satellite ground systems are today. Hardware centric, very siloed, very proprietary, 3GPP brought standards that all the carriers subscribed to. Really what that allowed them to do was to drive things like roaming, right? You'll notice your phone just works anywhere you go in the world. That's not an accident. That's done by these global standards that have been driven for years by the carriers and by the wireless operators. When you translate that to a government use case the need really isn't any different, any given military, they don't know where their next mission is going to be, but I'll tell you what, when they get there and they're in theater, especially if they're getting shot up, they need their connection to work. Just like we want our cell phone to work when we land in another country. Ultimately what will drive that is a very common philosophy around standards.

John Gilroy: Now, earlier in the interview, you mentioned the word container, and I remember seven, eight years ago, containers just coming in the market and people were amazed and astonished what they can do.Now it's kind of like yeah it's a container. That's what the Vim is all about. Maybe using containers in new and creative ways. Containers themselves have evolved in the last few years here. What about OpenSpace? How do you see the OpenSpace platform evolving to meet these changing needs? In other words, what is the high level roadmap for the OpenSpace platform?

Greg Quiggle: I think you'll see us after our one data release here really focused on three vectors of development. The first would be just broadening the portfolio of virtual network functions. Right? If you look at the number of specific hardware devices that sit in a teleport, if you were to walk into a teleport today, it's not like we're going to be able to provide all of those in the first few months of OpenSpace being an active product. What you'll see us do over the next several years is we'll prioritize those devices that we think would be most valuable to

virtualize. And we'll expand our portfolio to support that. In doing that what that allows us to do is then take the platform and support really a broader range of use cases, or as I've used the term already service chains.

Greg Quiggle: An example of that would be at first release, we're focused on EO sensing downlinks. You'll see other releases where we'll focus on satellite command and control links. We'll focus on two-way Satcom systems. We'll focus on spectrum management systems and really the way that you do that it's all based on this common platform, but to do it, you actually need a fuller portfolio of VNFs. In some cases, those will be Kratos developed, hence in the name “Open”. In some cases, those will not be Kratos developed. There'll be developed by an ecosystem of partners and in doing so, we can really increase the rate of innovation. The third vector, I would tell you that's really critical for us is broadening our base of supported Vims and orchestration frameworks. When I say orchestration frameworks, this could be service orchestrators, and there's many of them out there that are heavily utilized in the wireless and in the terrestrial communication to industries, an example would be like an ONAP, open network automation platform, which was sponsored by AT&T and Linux Foundation.

You have a private systems like Cloudify, and then you have standards that are in place that try to bring common functionality around orchestration frameworks, like MEF LSO. So you'll see us really broaden our set of certified and tested interfaces with those orchestration frameworks. That's what drives the automated dynamic nature of OpenSpace. On the Vim side, that's underneath our system. You mentioned containers right along with containers you have virtual machines, and these are the basic tools that allow you to run in a cloud-native or a virtualized environment. Just like initially you see us talk about platforms like Azure, there are unique Vims for other virtualized environments and or cloud native environments. There's unique ones for, for Google cloud, for AWS, for OpenStack, for VMware. As we get greater adoption with customers and, see more common pools of support for these basic virtualized infrastructure is you'll see us broaden our portfolio of Vim interfaces that we certify against.

John Gilroy: Crystal ball time. We've mentioned all kinds of things in this podcast. I mean, if you went to the dictionary, you'd get 50 or 60 new words and everything from ONAP to MEF LSO. All kinds of different terms, let's put those terms in a big pot and boil it for five years. Where are things going in the next five or 10 years? I mean, where's it all heading towards what's the ground station going to look like five years from now?

Greg Quiggle: The first thing I will tell you that I just find fascinating is for years and years and years in this industry, ground station locations were typically picked based on optimal coverage for the satellites.

John Gilroy: Of course.

Greg Quiggle: GEO satellites typically operate in the GEO belt as you would expect. There's very, very well-defined geographic areas where you typically see teleports. What I find fascinating is we already see this movement where the better location for the teleport antenna isn't necessarily for optimal satellite coverage. It's actually for optimal cloud infrastructure access. In doing that, you see this trade-off starting to happen, or an operator will actually make a very educated decision around sacrificing spectral efficiency or performance over air with the CapEx and OpEx needed to run their network over the life cycle. I think saying that even just a few years ago would have been heresy in our industry.

I find that fascinating. That's a really interesting balance that's happening. Along with that, I think some other things that are probably maybe less, less compelling. Today, I mentioned multi-constellation networks. You have LEO networks, MEO networks, GEO networks and in many cases, the ground systems are specific to those constellations, those platforms. What I think you're going to see here in the not too distant future are common ground platforms that are expected to adapt to these multi-constellation network. If a customer needs a very low latency service, perhaps they use a LEO satellite because it's much closer to the Earth's surface. If they need video broadcast with very wide range coverage, perhaps they use a GEO satellite because it covers more surface area on the earth, but it comes at the expense of things like latency. What you'll see today versus tomorrow is networks that are built to be able to on demand, leverage these different levels of constellations based on different SLAs.

In doing that, then the system can really react to the changes in user demand, but also can react to changes in the payload that they happen to be operating with an optimized for the payload. Last but not least what I sincerely hope is for the good of our industry, that we see really better in broad scale adoption of these common standards and interfaces that have been wild successes in our peer industries, terrestrial communications and wireless. We have to stop looking at our industry as special and it needs to be different. It really doesn't. You can see a huge parallel between the way a software to find payload works on a new satellite and the way a 5G base station works. What we need to do as an industry has recognize, that come together and drive standards, just like we saw with the wireless industry back in the days of 2G as they moved to 3G.

John Gilroy: Well, I'm glad you brought up the service level agreement SLA. I think that's going to be the key concept here, moving forward. I think different companies all over the world are going to have more options at their disposal of how much they want to pay for what, and you could almost customize a solution for that particular customer couldn't you given the requirements. It's going to be more of understanding all the nomenclature and the technology to come up with

almost a custom solution for each customer. You can do it now with all the software choices.

Greg Quiggle: Well, the coolest thing about it is it really happens dynamically, right? That's what SDN does for you. I'll give you one really easy, quick example, right? Let's say you have a service that's very latency sensitive versus a service that isn't. When you define a service chain for that customer, as a part of the SLA, just defining the latency sensitivity will actually dictate the way that it gets orchestrated in the network. You might throw more compute cycles at it to process the package faster. You might use a different framing, a shorter framing format, no minimize things like jitter. In doing that, what you're doing is you're getting latency performance at the expense of using more compute for that specific service chain. Usually that translates then to something the customer's willing to pay a little more for, right? This is all about value.

I mean, a case where the customer doesn't have latency, sensitive traffic, maybe they buffer effectively on both sides of the network. They don't care. You can offer them the same exact link with less stringent requirements around latency. You can run the same link on less compute. You see this really interesting trade off where over time, you're right it feels like custom services and in today's industry, that's custom services with the network engineer doing the job, designing it, deploying it over a period of days to weeks. What happens with software-defined networking is it's built into the basic SLA of the service chain, the business logic of the controller and it just works.

John Gilroy: To go back to the money question, what can happen is that today for a bespoke, a custom system would cost you a whole lot more, but with software virtualization, it almost allows you to customize it at a whole lot less expensive today. Isn't it?

Greg Quiggle: Absolutely. Absolutely.

John Gilroy: Well Greg, thanks for the insider perspective on ground station innovation. I would like to thank our guest, Greg Quiggle, Vice President of Product Management at Kratos.

Greg Quiggle: Thanks, John. Take care.