Constellations Podcast

Episode 38 – Open Source, Smallsat Software and Eating Space

Speaker: Marshall Culpepper, CEO and CTO of Kubos – 24 minutes

John Gilroy: Welcome to Constellations, the podcast from Kratos. My name is John Gilroy and I will be your moderator today. We are recording this podcast from the floor of the Small Sat Conference in beautiful downtown Logan, Utah, a huge metropolis, mountains, we got everything here. The smallsat market continues to grow rapidly. You can tell by looking around this conference with all these people floating around here. Mostly I think it's spurred by venture capital investments and by technical innovation.

Over 300 small satellites were launched in 2017 and 679 could be launched by 2022 according to Space Works Engineering. Today we are talking to a person at the heart of this industry transformation that is helping to make this vision, the smallsat vision, a reality. Marshall Culpepper is the CTO and CEO of Kubos, a Venture-backed start up that is building the first comprehensive software platform for the satellite industry. He is a software engineer and technology entrepreneur with over 15 years of experience. Marshall is focused on software and how an open source community and the Cloud can help smallsat mission developers.

Well, let's talk about the bell of the ball here. Hardware, you know? It looks like hardware is getting all the accolades and everyone's got a new hardware product here and there. Software doesn't get as much attention, kind of like the also-ran always a bridesmaid never a bride, you know? What role do you believe software should play from a smallsat perspective?

Marshall Culpepper: Well, the reality in the satellite world, just like any other area of computing is that, a majority of the hardware here is run by processors. Those processors require software. That software is an important part of the ecosystem here. I don't think that software is not important the space industry, in fact I think it's very important. What you see traditionally is that software in this industry has been sort of framed as a product that hardware companies manufacture. What we're trying to do just a little bit different is provide an open alternative that allows different hardware manufacturers to work together in a more bespoke manner. Instead of having to integrate every specific piece of hardware together for every mission, make it simpler for satellite companies to focus on their payload. That's kind of the core of our idea.

John Gilroy: So I was walking through Georgetown and I saw this tailor that said Bespoke Tailoring. I went home and talked to my wife and said, "Is that a brand name?"

She said, "No, you idiot. It's custom-made, but very exquisitely tailored custom-made and very expensive clothing."

Marshall Culpepper: Right.

John Gilroy: So that's what bespoke is. You just used the word bespoke.

Marshall Culpepper: Yeah, yeah.

John Gilroy: So that's what software could be and traditionally it was: “Okay I'm going to make this one package that's going to work for you for this one instance and adding or removing changes is going to cost you a lot of money”. So bespoke is expensive.

Marshall Culpepper: Right, it's expensive and it's very, very hard to replicate. The model with smallsats is most people are aware of that are in this industry at least, is that you're trying to not just create one satellite, you're trying to create 50, or 100, or 1,000. In order to do that at scale, not only do you need to be able to manufacture at scale, but you also need to have software that's reusable amongst that constellation of satellites. It just doesn't work if you have custom software for every single one.

John Gilroy: Believe it or not, I do preparation for these interviews. No one believes it. So I went to LinkedIn and typed in your name. I thought it was going to be Vice President or entrepreneur or software developer. Here's what comes up, "Software is eating space." Is this a movie title or something? It's in production in LA right now. So, "Software is eating space." What does this mean?

Marshall Culpepper: So Mark Andreeson, I think popularized this term that software is eating the world. It was kind of a famous phrase that sort of captured the moment in time which was the Silicon Valley bubble, I guess if you will. While I basically took that to the next step which is like, obviously software is the core and the heart of everything that is done in the space industry. People just have never really acknowledged that. Whether that's flight software on board, the mission software that runs your mission, the software you use to design and plan your mission. It's at every step. More and more of what we're seeing downstream, even from the satellite itself, is analytics and artificial intelligence software being used to monetize the data coming down from these smallsat missions.

John Gilroy: You're kind of reflecting the changes in the industry. So old fellas like me, old gray hairs here. You know, you put that satellite up there; it's as big as a refrigerator.

Marshall Culpepper: I have a gray hair too.

John Gilroy: It's as big as a refrigerator and we're going to have Marshall working that project for three years. Then we hope to make the date. Then we're going to do it all over again. Well, you can't do that with 50 satellites. You physically can't.

Marshall Culpepper: Yeah, right and it's not just about one operator doing it over and over for their own satellites. It's not every operator at every satellite. So the more constellations we have, the more unit economics we can, as an industry, we can realize. Therefore the more we can get out of these small satellites.

John Gilroy: The secret about custom software, in case you haven't figured it out is, humans put together custom software and humans can make mistakes.

Marshall Culpepper: Right.

John Gilroy: Version two or three may have different mistakes. If you can come up with some kind of a framework that other people can take a look at, other editors, editors around the world could take a look at and say, "Hey Marshall, you screwed up there again. Let's change this." This is called the open source community, where you take up concepts and ideas and just let everyone take a look and take a whack at it and, you know ... Guess what, you get progress that way.

Marshall Culpepper: That's right. Yeah, so my background specifically is in commercializing open source technology. One thing that open source has really taught the software industry in general, even much further than before the space industry. I think everyone learned this lesson that not only can you collaborate with your competition, you can both mutually benefit from that collaboration. For example, everyone probably at this conference has like, an Android or an iPhone device running a web browser. Those web browsers share the same source code at the end of the day. They have some differences amongst them, obviously, but Google, and Apple, and Microsoft and quite a few other companies all work on an open source project called WebKit that essentially runs the majority of all web browser engines in the world.

John Gilroy: You're pulling the curtains back! You're pulling the curtain back! Who's the man behind those curtains?

Marshall Culpepper: Right, right.

John Gilroy: It's WebKit!

Marshall Culpepper: It's WebKit, yeah. You know, the lesson learned there is that, friendly competitors can collaborate and can have a baseline. So, it's really common in this industry in the traditional sort of Silicon Valley, I guess, software industry, to

take that open source technology and to commercialize it. So, that's all we're doing. We're taking an open source idea, which is operating system around satellites, getting tons of people to use it, and collaborating on that source code and contributing back to it, and then commercializing that technology.

John Gilroy: Went to your website, fired up my browser, with all that open source code no one talks about.

Marshall Culpepper: Right.

John Gilroy: This is a great line here, “De-risk your mission?” Give me some of the- back that truck up. How do I de-risk my mission? So how can open source software de-risk my mission?

Marshall Culpepper: So most people, when they think of security, they think of, “Okay, I don't want anyone to know how this works and that's the only way that my thing will actually be secure.” But the reality is, if no one knows how your things works, then neither do you. And that means that as soon as there's a bug found or a critical security bug found in that implementation, you have no recourse. Because guess what? You don't know how it works, therefore you can't change it, you can't fix it. In open source, there's a sort of a “wisdom of the crowd” attribute to it. Anyone can see exactly how it works. So while yes, it may be easier to find a security bug, it is also just as easy to fix that security bug. So that means there's sort of war of attrition that happens naturally around any open source projects, around people trying to break it, and people trying to fix it. There's a much faster iteration cycle, and a much more secure environment over all. Basically the most secure environments in the world are all open source, is a result of that.

John Gilroy: And then let's jump right on NASA here. If you go to NASA and look at some of their projects, one of their fastest systems are running on...

Marshall Culpepper: Open source.

John Gilroy: Right! Right! Because it's fast, why is it fast?

Marshall Culpepper: They usually have all kinds of open source projects.

John Gilroy: People all over, people from Texas, people in who knows where, Ethiopia, that would get to look at it and go, "Well, why don't you do this?" I never thought of that! Bang! Quicker, faster!

Marshall Culpepper: Yeah, some of the earliest open source projects were actually created by JPL engineers.

John Gilroy: Yeah, makes sense to me.

Marshall Culpepper: Yup.

John Gilroy: So what do you think are the toughest challenges faced by smallsat developers when they're building their missions?

Marshall Culpepper: Oh, all over the map. I think some of the biggest concerns obviously are going to be around the integration of sub-systems and the integration with their launch vehicle/deployer. So what you can see is a pattern in most smallsat missions, is that they'll maybe allocate six months to a year to go from concept to building a satellite and actually handing over freight hardware to launch. And then they'll spend another six months to a year, in that time not realizing how hard it was going to be to write the software to integrate all those different bespoke components, and then also how long it was going to take to integrate directly with their deployer and all the physical constraints that come along with that. Basically, satellite companies don't- at least small satellite companies haven't learned the lesson that integration early is super important.

John Gilroy: I've been playing around with HTML and every time I go to a website, I go, “Wow! Michelangelo!” I mean, really complex things are being done. But most people, I imagine if you went to a 7-Eleven and grabbed 10 people, they'd say, “Oh, you're doing two or three things, it's pretty easy. There's no creativity involved, and there are no challenges there.” Why do you think so many companies underestimate your skillset?

Marshall Culpepper: Yeah, that's a really great question. The truth is, the software really is very artistic and in fact there's quite a lot academically written about the artistic nature of computer software. That being said, the thing that's interesting about software that most people don't seem to understand- who aren't engineers: is that it's just as much an arbitrary art form as it is like a form of engineering. So, while there are well understood, like axioms around how to structure code or how to make it optimal, there are also very specific things about software that are up to the eye of the beholder, if you will. So it makes it a lot more freeform and harder to wrap your head around as a concept, or even to try and estimate how long something might take to get done because to some degree it has a lot to do with how that person decides to tackle the problem. Which may not always be right or wrong, it just may be a certain way.

John Gilroy: I was doing an interview in Washington D.C. for a software company. I went to their website and I looked at their HTML code.

Marshall Culpepper: Right.

John Gilroy: And in the code it said, “If you're reading this, we want to hire you.” And that's the whole idea. It's almost light-hearted, but serious at the same time. It's a strange skillset that very few people have. But most people do want their own software for smallsat launches. Why would they even consider Kubos?

Marshall Culpepper: That's a great question. Right now the primary thing that people really want to control in their satellite is their payload. They want to measure some- if it's observation with a camera, or they want to communicate some specific kind of data with a radio. Those things in fact are the things we want to stay custom. So for us, what we focus on is all the stuff that they don't want to write. Which is like, the housekeeping of the satellite or the attitude of the satellite, or communications with the ground station. Like those things should just be figured out. The companies that are innovating with new technology and new payloads should be able to focus on just what their payload does and how to make that optimal. So for us, that's what we focus on. We make that stuff easy for them to do, but let them do it. And then everything else is handed for them.

John Gilroy: In Washington D.C. about six months ago I walked into a Microsoft developer’s user group, and their guest speaker was a guy from Red Hat. Which I just start beating my head in the wall, because I remember the guy screaming and throwing chairs at Microsoft and everything else. So I'm going to use this phrase and don’t you start screaming and running out of the room. But are you trying to become the Microsoft Windows for smallsats? Are you?

Marshall Culpepper: Yes. There's a few ways you can take that analogy. The Microsoft for the smallsat industry-

John Gilroy: For today's Microsoft, yeah.

Marshall Culpepper: Or the Android for the smallsat industry. Yes, today's Microsoft is a really great way to put that. That's right, yeah. We envision ourselves as the end-to-end software platform for satellites. And to me satellites are just another computing platform, just like mobile devices, just like PC's.

John Gilroy: How would your solution compare to traditional approaches in the geo market?

Marshall Culpepper: There are a few different options in the geo- like the larger kind of traditional geostationary satellite market, a few operating systems like the Ex Works and a few others that have been traditionally vetted and used for mission-critical applications both, not in just aerospace, but in automotive and other markets as well. Those operating systems, they have a really great track record of producing reliable missions and reliable code. The thing that they don't do, that we do, is they don't really focus on the hyper-vertical aspect of space flight. So, in other words the Ex Works, or a few of the other options in that broad

category, they're really focusing on the core operating system attributes. So they'll give you a general purposed scheduler and a few other handfuls of things. But if you want to learn how to pitch and yaw your satellite, well good luck. You're going to have to go write that mission-critical code on your own. They don't really focus on the really specific problems that satellites have, and that's what we're doing that's different from them.

John Gilroy: The pitch and yaw, definitely what an astronaut would use. But I think we walk around the conference and all of these booths, there's a lot of pitching going on.

Marshall Culpepper: Yes, there is. And I've done my fair share, too.

John Gilroy: In our world- in the satellite world, there's this concept of operation. It often takes time to plan out, during the hardware delivery and development stages. Can you accelerate that? Can that be accelerated?

Marshall Culpepper: Yes it can. In fact, one of the hardest parts- especially for a new team, understanding the operations of a smallsat is really just like what we call con ops or like the actual sort of nitty-gritty of how operations actually occurs and how you know how to prioritize the next thing the satellite does and those sort of details. So, Major Tom, our mission operation software- yes, is a valuate product you can purchase on top of Kubos. And that software can be used today, as soon as you start using Kubos in your platform, you can start using that in the lab, and you can start basically letting your operations team get practice operating your satellite before it even goes to space. And that makes the transition that much smoother for your team.

John Gilroy: So how are companies such as yours leveraging IT concepts? We talked about open source, another concept that's popular, the Cloud. How are you using that to benefit your customers?

Marshall Culpepper: So the Cloud is a huge aspect of our infrastructure specifically with Major Tom. With the Cloud, what that enables our customers to do is operate their mission from wherever they are. So they don't have to necessarily care about where their ground station is, where their ground station infrastructure is located. They can log on to a cellphone, they can operate their mission; they can use their laptop from Wi-Fi. And those things allow your operational outlook to be much more flexible and simple. And so, the Cloud is not just a way to store data but it's also a way to virtualize execution of the operations of your satellite.

John Gilroy: If you took a piece of paper and walk around the event here, you find all kinds of different smallsat missions, all kinds. I can't even have a conception of all the different ones that are out there.

Marshall Culpepper: And many of them are my customers.

John Gilroy: Yeah! So the question is that, if everything is unique, if everything is bespoke, then what about the off-the-rack what about the COTS? I mean, how can they tailor a COTS for their needs?

Marshall Culpepper: If you look at any of the smallsat missions here at the conference or just in general, the vast majority of them are using COTS to some degree, many of us. So, they're either using like an ISIS or GomSpace or Cloudspace, Pumpkin, any of these sort of off the shelf manufacturers, for some of their subsystems if not all of them. So, the majority of the subsystems that are COTS are the ones again, those are the things that pretty much any satellite needs to do, whether that's like, manage the charge state of a battery, or the solar panels on that satellite or even something as simple as just controlling the attitude of the satellite. Those things are common no matter what your satellite does. So those COTS boards are pretty much applicable across the board.

John Gilroy: Great. From the brief 15 years in the business, got a startup; so what have been the biggest challenges of being a startup company in this smallsat market?

Marshall Culpepper: Because all of our customers are satellite companies, and a lot of them have launch delays. I have the unfortunate marriage of being- my customers waiting on launch delays even though I'm a software company. So I have some of the challenges of a hardware company waiting on launch, even though we're just software.

John Gilroy: Could be the worst of both worlds.

Marshall Culpepper: Yeah, sometimes it is extra challenging. But in other ways, it's a challenge that I fully welcome, because we're not going to get there and virtualize satellite infrastructure without a common software layer. That's my ultimate goal.

John Gilroy: So you have 3000 kindred spirits here. What about the kindred spirits who couldn't afford to come here? There are kindred spirits sitting in debt in Texas, studying math at the University of Texas, a great school in Austin there. You know, maybe they're engineers thinking about where should they take their EE skills. Where should they take their mechanical engineering skills, or their software development skills? What advice would you give them for developing smallsat and getting it online?

Marshall Culpepper: Yeah, absolutely. So one of the beautiful things about open source and one of the reasons we like to do it is because it gets people in the door much easier, much quicker. So we actually have about four or five different university programs currently using our open source software, and they aren't customers

yet, but they will be when they graduate and go start a job, "We've already been using your software for the past several years."

John Gilroy: I call that the Tableau model, that's the Tableau-

Marshall Culpepper: It's just literally the Tableau model. And actually Microsoft did it for a while too; they gave away licenses to their software for free to university students to kind of get them hooked on the software. But for us the open source model means, not just commercial or reliability, commercial re-usability, it also means getting people in the door at a much faster pace.

John Gilroy: Well we're doing a podcast here so people can't see what I'm holding up. It's actually a solid gold crystal ball. It's a crystal ball, and you've got to peer into the ball. Peer into the crystal ball, and tell me; five years from now, I mean how can you predict five minutes in this business? I'm asking, five years down the road, smallsat industry, where do you see the software in five years?

Marshall Culpepper: Well I think, it's inevitable that more and more satellites will have to be more and more focused on software. So what we see today already, even outside of Kubos is more and more satellites are starting to adopt the idea of being responsive, or virtual in their software capacity. In other words, being able to reconfigure what the satellite can do with software on demands. And I think five, maybe ten years down the road what we'll see is- well today what do we have with servers in hardware rooms? Well, we can allocate them on demand with software. We can say, let's call the Cloud. I can say I want that chunk of that hardware now. That's what'll happen in the satellite industry I think, five, ten years from now, is we'll be able to virtually allocate capacity of both sensors and communications capacity using their software. Then we'll probably have to launch less missions because we'll be able to reuse what's already out there.

John Gilroy: Sometimes I have coffee with friends of mine who are software developers, and I'll ask you the tough question, you know?

Marshall Culpepper: Yeah.

John Gilroy: There are libraries of open source code out there. Is a software developer going to be some guy who knows how to use the libraries well, and cut and patch and put chunks of software together? It seems like it's a chunking process rather than a carrot and close the carrot. It's developed to be more modular and chunk oriented. How's that for a phrase?

Marshall Culpepper: No, you're right. I mean, I would say a software developer's primary job is integration of libraries, applications, business concerns, it's integration. That's what it is.

John Gilroy: And the libraries have all been tested, and vetted through open source so-

Marshall Culpepper: That's right.

John Gilroy: You'd not be taking a chance.

Marshall Culpepper: Yeah, I mean it's not like you're not writing any new code, but you're writing enough that you have to integrate with lots of different chunks that like you said, it becomes a full-time job.

John Gilroy: Well, unfortunately, Marshall, we're running out of time here. I'd like to thank our guest, Marshall Culpepper, CEO and founder of Kubos Corporation.