

NETEVENTS

GLOBAL PRESS & ANALYST SUMMIT
INNOVATORS IN CLOUD, IOT, AI & SECURITY

DRAFT

*Keynote Presentation - Mansour Karam, CEO & Founder,
Apstra*

"A Matter of Intent"

Mansour Karam

Excellent. Good morning everyone. Very exciting to be here. Very excited to be here. A great night yesterday. We had a lot of fun. Thank you for all the recognition. We're very humbled. A really great event and great to see that it's going to a good cause as well. Before I start, I just wanted to ask a few questions. How many of you guys cover IoT? Yeah, many of you. How many of you guys cover machine learning? Okay, quite a few of you guys. How about networking? A few of you. Well, what I wanted - if there is one takeaway from today's presentation, is that all of these technologies are deeply interconnected and related, and that none of - neither one works without the other. Let's get started.

Let's imagine a day, an ordinary day in 2025. It's Monday morning. You wake up out of your bed. Your coffee is ready there for you in your smart home. You get into your car. Your car knows where you want to go. It knows it, because it's on your calendar. Essentially, you're going to work. It knows it's Monday and you're probably in a grumpy mood and, therefore, instead of showing you 101, which, if you're in the Bay area, you know 101 is not the nicest site out there. It changes the scenery through virtual reality. You're now south of Italy on a back road. Maybe changes the soundtrack. You have a nice V12 roar in the back, in the background. It knows you have a video conference, so it starts it for you, a high-resolution screen. Through augmented reality you have all of your data, the information you need to be productive, at your fingertips.

Maybe even including the mood of the participants, maybe pretty much everyone is grumpy. It's Monday morning and, therefore, that should guide your tone. This, to me, is clear. We're getting there. It's kind of line of sight, based on all of the technologies that are available today, and that we're perfecting in the next 10 years. You wonder what all these technologies are. What makes it all work? Well, there has been, really, a convergence of many technologies out there. The hyper scales, out of necessity over the last 10 years, really worked very hard perfecting them and making them available to the rest of us. I would say there are two categories. There are some proprietary application-specific technologies. For example, if you are Google, it's Search. If you're Amazon, it's Online Shopping. If it's Facebook, it would be Social Media. Then there is many, many technologies, that I would call infrastructure technologies, that were critical to make it all happen.

In fact, they're critical to the transformation that we call digital transformation. They're at the core of all of those technologies that are making - that have the potential of really changing our lives. All of these technologies are now, and they're available today and that's great. Now, but that's not all, right. Also, let's consider the number of endpoints joining the network. Today you have 11 billion IoT devices. By 2020, that number is going to 30 billion, and by 2025 that number is going to be 80 billion. In fact, it's estimated that 152,000 devices will be joining the network every minute. For every one of those devices that joins the network, essentially, they're going to be generating traffic that is disproportional to - that will increase disproportionately.

In fact, that devices, when it joins the network is going to connect to many other devices and many other users. If the increase in devices is exponential, the increase in traffic is actually doubly exponential. In fact, we're witnessing a network traffic explosion that has been - that we have never seen before. I was at a service provider event earlier this week and they were saying how traffic is increasing by 50 per cent year over year. Of course, those hyper scale companies have recognised the problem and for the last 10 years they've worked diligently with the vendor community in order to really build the type of foundational network that will support this trend, essentially, in terms of bandwidth scale out, the networks that can scale out. In terms of reliability, these networks need to be extremely reliable. Also, in terms of automation, the ability to operate these networks in a massively automated way at scale.

What about the rest of us? As a result, these companies, of course, have become leaders. They have gained market dominance and they have a competitive edge that is really hard to surmount. What about the rest of us? These technologies are now all available and, therefore, every CIO is very aware that they need to be part of this digital transformation. They need to embrace IoT, machine learning, artificial intelligence. In fact, CIOs know that there is no such thing as a traditional business today. You're either a digital business, or you can't compete. Think of businesses that are - you consider traditional, like an insurance company. Today, available to these insurance companies is IoT data from every car out there, every mobile phone.

In fact, they should be able to gather all this information and then analyse it through machine learning, and that should be able to guide their premiums [to their] customers. If they're not doing that, then their competitor will, and they will be unable to compete.

Every business needs to be transforming digitally and taking advantage of those technologies, but there is a little problem. The industry has a little secret that I'd like to share. Eighty-five per cent of networking teams today still operate their networks like they did in 1995, essentially, manually, box per box, using command lines which are [arcane] and specific to every hardware vendor. In fact, they operate their networks the way - in what is the equivalent of assembly language for those boxes.

1995 - how many of you remember Windows 95? In fact, that was the hottest operating system. In fact, it transformed the industry back in 1995. Let's see. I may find one of those, maybe, at one of the old libraries - public libraries out there, but this is quite extinct. Or this phone. That was the hottest phone in 1995 - Nokia. Does anyone remember Nokia phones? Actually, I came to the US in 1995. That's a picture of myself with my dad when we first visited Stanford when I got here. That was 22 years ago. Clearly, that was a long, long time ago, judging from this picture. The point is, 1995 is a really long time ago.

Trying to embrace digital transformation and machine learning, and be part of the IoT trend, yet operating your network like you did in 1995, is the equivalent of wanting to build the house of your dreams, yet being unwilling, or unable to invest in a proper foundation. Not a great idea. In fact, your digital initiatives will fail if you don't change how you operate your network. I'm not the only one saying it. Leading analysts out there are saying it and, certainly, CIOs are very aware of this. There is good news. As an industry, as a networking industry, we have been working very hard to solve this problem. Certainly, at Apstra, the company was founded back in 2014 to address exactly this gap.

Us, along with others in the industry, are really working to deliver on solutions to this problem. Of course, we live and breathe this every day. The product that we deliver is called AOS. Through delivering this product, which came to market last year, we learned a few lessons, which I'd like to share with you here. What's really needed is a self-operating network. It's a network that essentially delivers on autonomous operations. This means that the human factor needs to be taken out of how we operate our networks. The network needs to be able to self-configure to fix itself, to defend itself, to document itself. In order to achieve this, we've looked at approaches of automation out there, and we concluded that the critical attributes are intent based, closed loop and vendor agnostic. Let me describe these a bit for you.

Intent based is the notion that, as a user, the way you interface with the system is by describing your desired outcome, as opposed to describing the specifics of how you want to get it done. If I were to take a self-driving car analogy - and this analogy actually works quite well. A self-driving car is very sophisticated. The software needs to be very intelligent. In fact, I would submit that infrastructure is even harder. It has more components and it's even a harder problem. In the self-driving car, ideally, as a driver, you get in and you just tell the car where you want it to go. That's intent based. In fact, you may also have some opinions. You may say, well, actually, on my way to work I - if my laundry is ready, I'd like to pick up my laundry along the way. Your car should be able to include that constraint as part of its algorithm in terms of delivering on your intent. That's intent based.

Now, there has been approaches to automation in the past, but what was missing is this second attribute, which is closed loop. Again, if I were to think of a self-driving car, it will be impossible for this self-driving car to get you to your destination if it's not gathering telemetry of its environment. In fact, self-driving cars do. They have thousands of sensors, gathering thousands of parameters every second and the software is building a notion of the actual state. The car is aware of its environment. The software compares this actual state to the desired state, which is derived from your intent. Then, based on this comparison, the software controls the car to make the adjustments so that the car continues delivering on your intent, and driving you safely to your destination.

If the changes are outside the capabilities of the software, then the software will recognise this and will generate an anomaly and alert the driver to take the wheel again. That's closed loop. It's critical to delivering on an autonomous infrastructure. Then the third attribute here is vendor agnostic. If you're interfacing with your system by describing what you want, rather than the specifics of the commands for that specific vendor's hardware, then you have a major opportunity here to decouple your operational model from the specifics of the vendors underneath. Therefore, you have the ability to become hardware-vendor independent. Of course, every infrastructure out there, every data centre infrastructure out there comprises hardwares from multiple vendors, so becoming vendor independent is critical to delivering on this vision.

When you think of an autonomous - on the software delivering on autonomous operations, then, especially when you consider the intent and sliding scale of intent, have it being at the high level, but also providing the opportunity to describe your specific constraints and opinions, and the - gathering the telemetry from all of the devices in the network to continuously validate that your infrastructure is delivering per your intent, I think it should be clear that at the core of the problem is really a large state - distributed state management problem. Certainly, we've recognised that. The core of our system is a distributed data store that is a repository of all of the states in your infrastructure.

It's not only a matter of scaling out the collection of the state, but also it's - what is needed is representing the states in a way that captures all of the relationships between the physical environment, your virtual environments, your logical entities and all of the telemetry. Once you do that, and only once you do that, do you have an ability for the software to powerfully reason about the state and to provide the deep visibility to the operator as to what's going on in their network, which - both of which are critical to delivering on the vision of an autonomous infrastructure. Doing this properly, involves layering in all of the abstractions so that the majority of your code is vendor agnostic and only specific components - we call them device agents, very much like device drivers, which I think we're all familiar with, that are experts at the specific devices that they're controlling.

You would have a device agent for specific vendors' hardwares that are experts at extracting the capabilities from that hardware, essentially, either pushing the configurations or gathering the telemetry. Outside of those agents, the software itself is vendor agnostic. If you implement software such as this in your infrastructure, then

you'll get benefits that you really cannot do without. Massive improvements in agility, which really enables you to recoup the revenue from being able to deliver on your business services, on your initiatives, on time. Massive reduction in outages. We know that 80 per cent of outages are caused by human error. Massive reduction in operational expenses. We know that organisations spend \$3 to \$4 operating every dollar of CapEx and 80 per cent of it is manual - redundant manual operations. You can see how you can easily recoup your entire CapEx spend by implementing this technology.

Last, and not least, you can refocus your engineers. Instead of your engineers spending 80 per cent of their time running mundane tests manually, they can retrain, hone their skills to become instrumental to the areas of the business that are more strategic to you. It's been really great to see the excitement around this technology. Clearly, there is a massive gap out there. Since we launched last year, we've generated revenue globally. We're very pleased to see the largest vendors, the leaders in networking, jump on this as well. I'm very excited and I'm very optimistic that we're going to be solving this problem, starting today and in the next years ahead. Really, when you look at the skyscraper - I like this analogy - ultimately, you spend a lot more time building the foundation, and the foundation is the most critical.

I submit that the network is really at the foundation of everything else you're doing. We're working diligently to really build the right foundation for your business in order to enable you to deliver successfully on your digital initiatives, and so that you have the ability to compete effectively. Thank you.

Manek Dubash, MC

Mansour, thank you. Now I'd like to invite Scott to come down and interview you.

*Keynote Interview & QA -
Mansour Karam, CEO & Founder, Apstra and Scott
Raynovich, Principal Analyst, Futurion*

Scott Raynovich

If this were intent-based system, it would have known that I wanted to speak. Good morning, Mansour, very interesting speech and congratulations on the award last night, pretty exciting.

Mansour Karam

Thank you, yes, we're pretty excited.

Scott Raynovich

I like the vision of the car just taking me down the road, and soothing my nerves. I'm looking forward to that technology.

Mansour Karam

Me too.

Scott Raynovich

You talked about a lot of stuff there, your closed loop system as you described it. I'm wondering when you look at Apstra's enabling technologies, your IP, what are the key enabling technologies that are going to differentiate you going forward?

Mansour Karam

A great question. As I mentioned at the core of the networking problem is really a distributed state management problem at scale. I think that having that distributed systems expertise is critical, and it's a very hard problem, and we have the right skills and expertise in-house to deliver on that.

In addition, intent is really about building in the right abstractions, abstractions is an over-used term, and I think that my co-founder, Sasha, if I just describe it as an abstraction would think that I'm not describing it properly, because indeed it's a lot more complex than that, it's really the ability to describe intent at the various phases of the life cycle of your infrastructure. Having that experience, and having that knowledge of abstractions, and how to abstract out the system, I think is critical. We have that skillset in-house.

Number three is all of the things CCIEs know, network engineers, all of the knowledge that they have built up in their heads over the last 30 years, operating networks. You need to have those network engineers on staff, and you have to be able to - for your best software developers to partner with the network engineers to extract that knowledge from them, so you can implement them and automate those processes in your software.

Of course, last but not least, it's about having the best software developers. I think it's known in the industry that the difference between the best software developers and the not - and those that are not the best ones, is massive. For us we work extremely hard at really building a culture at Apstra that attracts those types of really top performers.

Scott Raynovich

Great. Let me ask an audience question real quick, before this event how many people had heard of Apstra? Okay, not bad.

Mansour Karam

That's great.

Scott Raynovich

How many people have heard of Cisco? Okay. This is something that came up yesterday with your shark tank panel, and it's also - I've written some blogs about this that have caused some - raised some hackles, the whole Cisco question. Then we have this marketing juxtaposition, which I thought you eloquently explained how you ship the product and then suddenly there they have intent based systems too, isn't that interesting. How do you think this is going to play out?

You obviously know that Cisco is going to come after you, they've got immense resources, how are you going to negotiate the little start up battling the giant?

Mansour Karam

I think that ultimately - well, first of all, I think we have just from my experience, this is not the first time we have an idea and we come in and we say, we're going to compete against the big guys. I think that ultimately start-ups, that's what they do, that's what we do for a living. In fact, the trick is to really hire the best people, so that as a small team you can deliver more than any other team. The trick there is to build the right foundation, and to invest in building the right foundation, but then also to do one thing better than anybody else. It's really important as a start-up that you focus on one problem, and for us it's the data centre network.

Now having said that, specifically as it relates to Cisco, we're actually - what we said when we started the company is that we addressed - we said we were going to address the user problem. As I mentioned, customers out there today know a few things, they want hardware from multiple vendors, and it is very clear. I think the number is 80 per cent of organisations out there want to have a dual-vendor strategy, that's number one. Number two is that they know they need to automate, and therefore they need APIs.

These are the two foundational I think components that to us indicated in 2014 that there was an opportunity for Apstra to get in and really change or to [disrupt] - change the value proposition in the industry. Ultimately, I like to say the customer indicates true north, they're the ones that tell us what to do, and as long as we're focusing on delivering on the customer problem, I think we'll do well. Of course, the partners will follow.

Many of those customers actually do have some gear from Cisco, from Arista, from Cumulus, Juniper, Dell, HP, and those vendors work with us harmoniously to deliver on those solutions for our customers.

Scott Raynovich

Of the others you named, are there other competitors that keep you up at night, other than Cisco, would you name them?

Mansour Karam

I'm a big fan of the saying that you run your own race. I think we run our own race, we deliver on our customer - on what our customers want us to solve for them. We work with our partners and other vendors out there to deliver on solutions for them.

To me ultimately as long as we're focusing on the customer, as a vendor community, I think there is a lot of opportunity here to partner and deliver on - more than the sum of the elements themselves.

Scott Raynovich

Apstra, you're a hot company, whatever that means. I think did Gardner call you hot company, I called you a hot company, Michael called you a hot company, right. You have a unique position with a track record and I believe you guys are self-funded, correct?

Mansour Karam

Correct.

Scott Raynovich

You didn't have to go raise capital from - suck up to VCs, which nobody likes to do. But every start-up in its life has some very - a crisis, or a moment of test. Have you reached that point yet, do you see it coming, and how do you prepare your team for that?

Mansour Karam

This is a great question. I always say that having started my company maybe 10, 15 years after I started, gave me all that experience through in the three other start-ups that I thought were so instrumental for my ability to really run this one, and start this company. I think one thing you learn is that challenges are always going to be there, but that really as long as you focus on your goal and on your destination, ultimately you will have the built-in ability to ride them out.

Certainly, we have faced - I'm not going to call them crises by any means, by certainly there is always situations where something happened that is not ideal, and that you just have to ride out. I would say that we've been at it for more than three years, so it would not - certainly we have faced a few of those.

Scott Raynovich

Maybe you're trying to build an intent-based culture, is that a bad thing?

Mansour Karam

Actually, that's a really good point, I didn't think of it that way. But yes, ultimately our culture really defines rules in kind of a framework of how - what we expect in terms of behaviour, as opposed to describing specifically the how. Then we try and have every member of the team feel and have - feel those rules, or these guidelines, in every cell

of their bodies, so that they can themselves lead using those same culture elements. I guess that's intent-based.

Scott Raynovich

Excellent, thank you. Some other questions came up yesterday in the shark tank especially, they were drilling you about your customers and we all know people don't want to talk about their customers, or disclosure names. Can you tell us anything else today, maybe describe some customers in more detail? Or where you think this is going to go in the next year?

Mansour Karam

I'll say two things there, is that infrastructure software first of all takes time, so as I mentioned we spent a lot of time building the foundation. This year is when we're really taking these customers to production. We're working with some of the biggest names out there, I think that's customers who are early adopters of technology, they're not the ones that want to go out there, and not the ones that are eager to go and talk about it. Some of these names we may never have as public reference.

At the same time, I think there is a lot to come, we expect good things to happen in the next few months and years, and I would just say stay tuned.

Scott Raynovich

Excellent. Well that's a great place to stop, I'm done with my questions. Thank you very much this morning, good luck.

Mansour Karam

Thank you [applause].

Scott Raynovich

I guess we'll take questions here.

Male speaker

If you could describe a bit of the infrastructure you need that like on say a Telcore to be able to implement for your system. Are we talking about major investment in servers to operate it, or beyond the agents and such, how much does it actually - what's needed on the networks' carrier side to implement?

Mansour Karam

Actually, so the way - this is software and it runs on virtual appliances either VMware based, or other. It's a matter of how big your infrastructure is, as I mentioned we're collecting a lot of telemetry, and we're representing the state in your infrastructure. But the technology itself scales out, and so you have the ability to just add more virtual appliances as proportionally to the size of your infrastructure.

Generally, I don't see that - we're taking an approach that I would say is not necessarily too heavy weight. For example, we collect the telemetry that we need to collect in order to validate intents, there are approaches out there that just collect mountains and mountains of telemetry just for the sake of collecting them, and then have the user try and sort it out. We try to take a more efficient approach there, again because ultimately state management becomes a bit more challenging problem.

Male speaker

Hi Mansour. I've been trying to figure out closed loop, and it's a key part of your message. When I look at closed loop it's here's intent, there's instantiation of the intent in the network, there's operation in the network you have to do some telemetry, and then analyse and then throw in ML and AI in at that step. Now you don't close loop back at the top of intent, there's all these pieces and parts in the network and it seems to me closed loops have to happen in every part of the network, or whatever pathway is instantiated to deliver on the intent.

Can you say more about how many closed loops? Are there big ones, small ones, tiny ones?

Mansour Karam

That's a great observation. Closed loop is not straightforward, ultimately closed loop is the idea that you're gathering the telemetry that you need to - so that you become confident that you're delivering on your intent. Depending on - ultimately when you're think for example of test driven development, to me this is a good analogy. You're asking your software engineers to write tests, and these tests are always running so that if you let's say create a bug, you will be capturing it.

Ultimately the most tests you have, the more relevant your tests are, the more confident you become. It's very similar when you're closing the loop, where autonomous infrastructure operations. There is certain telemetry that you want to gather, and if you can gather it, maybe sometimes it's not even network telemetry, maybe it's your application telemetry, because ultimately that's what you want.

Your system needs to have the ability to gather this telemetry in an extensible way. Then you as a user should have the ability to augment those tests in an environment that is particular to you, so that you build your own confidence that indeed you've closed the loop effectively, and that your system is delivering on your intent.

These are some of the aspects that we're really thought through in terms of how you collect the telemetry, how you extend the collection of the telemetry, how you aggregate that telemetry, how you reason about this telemetry across different sets of telemetry in order to ultimately get you the answer that you want.

Male speaker

Can you give an example?

Mansour Karam

Yeah, actually a great example is - maybe I'll refer you to a public document, which is a research paper by Microsoft Azure. I don't exactly remember the title but it was something about great failures. It was a really insightful paper that described how most problems in networks are actually great failures. Black and white problems are easy, if your interface goes down, you know it's down, and generally your product will be trying to avoid that interface.

Generally, that's not what happens in networks, what happens in networks is that a switch becomes a bit ill, so packets that go through it for some reason are either getting dropped, or having high latency once in a while. Generally, you can boil it down to transceivers maybe, your transceivers aren't seated exactly properly. Protocols really don't catch that, and in fact the more redundancy you add in your network, the more - the bigger your problem becomes, because a subset of your traffic will always go through that switch which is currently ill. That's affecting your applications.

How do you notice that, and how do you notice - how do you describe the intent, and how do you notice that indeed your network is not delivering on your intent? Ultimately, it's really about you have to take a probabilistic approach, you have to say, well I'm going to collect telemetry across my network, they use [ping mash], but that's essentially the type of telemetry that we could collect with our system. You conduct analysis that shows that, wow, I'm experiencing some packet loss.

Then you collect telemetry from your applications, and then by comparing the telemetry you're getting from your applications, and the telemetry you're collecting from your networks, you can deduce that indeed there is a problem, that the problem is because of the network and then that switch is where the problem is coming from.

It really requires representing the state properly, it requires extending the telemetry, having the ability to reason about the telemetry properly, and closing the loop.

Scott Raynovich

Great. I think we're out of time here, so thanks again, Mansour, thanks for the great questions.

Mansour Karam

Thank you.