

MARTIN Hello, I'm Martin Gasevski. I know many faces us over here actually, so thanks for showing up
GASEVSKI: on Thursday last class of last day. So, you know, A for effort for both of you. Yeah?

MICHAEL PARES: And Doug McLean is also here from Jitterbit. Thanks for showing up after lots of drinking on [? student's break. ?] Absolutely-- and if you guys have some drinks in here, that's OK too. I'm not going to tell anybody.

But you won't need it. You'll get excited about this just as much as I am about alcohol.

[INAUDIBLE] So these guys have really helped.

I've got to get some formalities out of the way first. This is a disclaimer from my company. If I didn't have this slide, then I wouldn't be allowed to be here to share information with you guys and talk to you guys, so-- you know, take a second to look it over. It just says that I'm speaking in a educational context, and not sponsoring or endorsing anybody. And then, Martin, you should talk about this one.

MARTIN Yes, I'll let you guys read this one. No, I'm not. So anything we'll say I guess can be used
GASEVSKI: against me, so please be patient and mindful that all of this-- maybe many of these things-- will be forward looking, stuff that are prediction. They are road map stuff that may not materialize in the future. So--

MICHAEL PARES: Great. So today, like I said, we're talking about Evented Web. You may also know it as Webhooks.

Microphone? Oh, is it not on? I had it on. I was asking everybody if they could hear me. Now it's probably too loud. Is that better? OK. Fantastic. So in the video, they won't hear any of that, but that's OK. I'll pretty much say everything over and over again.

So we're talking about Evented Web, also known as Webhooks. And what we're going to be talking about-- the learning objectives here-- essentially intergrating systems using only a user interface, being able to do quick prototyping, quickly sort of getting to yes inside of your company for particular integration. Because what Evented Web helps us do is abstract away costly integrations, makes it very simple to do integrations-- extending Evented Web using a middleware approach and doing things just more easily in the process engine.

So like I said, I'm from Greenpoint Technologies. Greenpoint Technologies is owned by Zodiak

Aerospace, so I've got some-- we used to be privately held, could do whatever we want. Now, we have some stringent requirements because we're publicly held company.

But what we do is VIP aircraft interiors. We're a VIP aircraft interior completion center. So what that means is we do luxury large private jets for billionaires, for governments, for heads of state, for corporate jets. And these are Boeing business jets. These are sort of the largest jets-- 747, 787, those kinds of things. We pimp out planes basically. That's not right. Pimp your plane, not pimp out plane.

OK so, we have several locations in the United States-- multiple in Washington state-- our headquarters in Kirkland, where it's really just information workers. Marysville, we have a prototype shop, a hangar facility in Moses Lake where we do modifications, cabinet shop, machine shop, and repair shop down in Denton, Texas.

You know we've evolved quite a quite a lot over the years. You know, in the beginning when I first started there-- 70 people-- we're now 400 people. We used to only do sort of the engineering and the project management and sort of brokering of parts. But now we do basically absolutely everything from the start of the program to the other program-- all the certification, all the testing, all the touch labor and everything. And the results can be pretty interesting, mind-blowing.

So like I said, you know, Greenpoint Technologies is a company of really high achievers. And when we're talking about sort of the requirements from our customers, you know, there's a huge focus obviously in the aerospace industry on delivery-- on delivering on time, delivering on your promise, and that level of accountability. So we always like to brag about some of our achievements in that manner. That's what Greenpoint is all about, delivering early-- like this Boeing business jet done in 2009 88 days early. It's unique in our industry. You can see some of the results of the work here.

We are also the first to deliver a 787 in 2016 for private use. And when we talk about the scale of money, the things that we're working with-- to obtain a 787 as a private individual, we're talking about \$200 million for just a bare, empty plane off the factory line. Another 100 million or so goes into an interior, like what we do. Our projects last anywhere from 18, 36 months, depending on the scope, and you know, depending on the configurations. We can often do sort of multiple mission configurations for these planes where seats will get taken out, more seats added, depending on the mission.

So we were the first to deliver in 2016. We're delivering a second 787 in 2017. That's what-- these some of these things are design concepts, renderings. I'm not always allowed to show off the actual pictures of customers' planes. But if you do some sort of Googling then you can find-- actually, there's lots of press and lots of stories. And you know, anything that's probably allowed to show, you can find a lot more out there. Even that stuff, I'm not necessarily allowed to show, but still some of these are pretty cool looking.

So Greenpoint also does some products, you know, rather repeatable products. Still not high volume right, we're talking about in my industry-- it's completely custom stuff almost all the time, you know, from beginning to end. No two planes are alike. No two customers are alike.

But in the last few years we actually started doing some products for overhead space utilization, like the arrow loft that you see here which are sleeping berths in the back of a 747 where there's some space sort of just above the seating. And these are these are modular. We've delivered multiple sets of these, multiple planes.

There's also somethings that we haven't delivered yet, because we haven't had enough orders. This elevator in a 747, for example, would take two orders for us to make money. And we've only had one, so we haven't done it yet. So we're looking for a second. If anybody in the room knows any billionaires or any-- I think we do only wire transfer and escrow. But you know, yeah, right, exactly-- I mean do you have a 747 that you need a elevator?

AUDIENCE: I'd love to help you out.

MICHAEL PARES: So this is the aerolift. And you know, I wanted to tell you guys a story about this, about sort of what the cost to implement is like. It's funny, over the last few days, you know, I've been thinking a lot about this presentation, the things I'm going to say, and realized that as people ask me questions, I end up telling sort of the same stories over again. And I'm like, you know, and this is probably what people might want to hear now.

So I was telling Charlie just I think last night, sort of an example of some the crazy things that Greenpoint gets asked to do for our planes. And there was one customer who had a bulletproof and bombproof S-Class Mercedes. And what they wanted to do was take that vehicle with them every where they went, so they always had it. They're always protected when they're driving around. So they want us to, you know, put in a hydraulic lift in the back of a plane-- cargo hold and strap it in, do all these things.

And one of our design guys as he was, you know, trying to just get the deal to close, and thinking a lot about it, realized that-- sort of kind of crunched some numbers and realized that-- it would be less expensive for this person to outfit 15 more of these S-Class Mercedes and store them in all the different cities that they're going to be flying to on a regular basis, then to have us to cut a hole in the bottom of the plane, and certify, and do all the things that are required to have him take it with him everywhere he went.

But that's kind of, you know, sort of crazy things that we get into. As you can see, you know, some of these requirements actually get into new products. And like I said, each one is unique. And the complexity around making all these requirements and all of these desires airworthy is sort of the point I'm trying to make. So--

I don't think they did. We didn't put a hole in the plane. I don't think they worried about the Mercedes either. Maybe they did, I don't really know. It's kind of funny, because, you know, there's sort of like a firewall between me and most of our employees and our customers. You know, we deal with a lot of customer reps. We don't get to find out a lot of information about what happens after the fact.

So that comes to this where I wanted to talk about what Greenpoint's sort of point of view is-- what our current state is, what our vision is for ourselves. And like I said, to manage that complexity, you need to find ways to sort of simplify ideas. You know, because there's so much communication, so many things like that.

So we like to put together a little charts like this. Last year we showed something very similar to this-- my counterpart [? Oran ?] did. And he was just talking about basically, you know, the way that we use systems and especially with Fusion Lifecycle. That, you know, it starts with process, starts with things like the Fusion Lifecycle workflow engine and being able to define and manage our process in Fusion Lifecycle, which it's so good at.

And from there, going to reporting-- extracting that data and having insights and, you know, being able to garner insights from those measures and things that we get from all that information, which then drives decisions. And in this example is about engineering management-- enabling them to make all the right decisions, knowing people are following the process, understanding where all the bottlenecks are from the reporting, things like that.

And then, just in general sort of, what has Greenpoint implemented? We started off with

engineering release management, change management, certification program management, and quality control. We've integrated with Vault. We control all of our drawings of our objects in Vault through PLM and Jitterbit-- or through Fusion Lifecycle and Jitterbit, sorry. And integrated to SQL Server SSRS for our reporting.

But there's a little bit of trouble for us with this, or at least a desire to do a little bit better, right? And that really comes down to speed. You know, when we're extracting data from Fusion Lifecycle, when we're trying to put these reports together to get people to make decisions, sort of the way that we end up having to do it so that we're not affecting the performance of people's work day is you have these things coming out extracting data in like nightly batches, right? So your reports are always a day old, right? And therefore, your decisions are always-- or sometimes-- a day behind, you know? And really what you want to have is instant decisions.

So I started think about, what would it take to speed that up? This isn't, you know, the only thing that I think about in terms Evented Web, but this is one of the main things that I think about. That in order for us to get to real time data and still have the integration, the reporting, the insights, and everything that we need, we need a way to sync things up that's immediate. And so this is sort of what our vision is now. This is where we want to get to. I think that the Evented Web-- the things that we're going be showing you-- is one of the best ways to get there.

But the projects that I have slated for this year that involve Fusion Lifecycle and this work where we're going to be introducing more budget and resource management, doing more document control processes, introducing-- you know, acquiring vendor data, and processing vendor data, managing the mod center, the installation, and the installation progress through Fusion Lifecycle.

And then, we need with that the ability to integrate-- do better with systems like Anaplan, Replicon, which-- Anaplan is like a sort of-- I don't know-- a web-based Excel thing on steroids, where you can make planning models, you know, resource forecasting, and budget forecasting, and all these kinds of things. So we want to extract our data from Fusion Lifecycle and send it to Anaplan.

Replicon is our timekeeping system. Being able to make sure that, you know, the moment that we have a new drawing added that we've got a new charge code in our timekeeping system for people to charge to. You know, all these things we have today, we have processes that

cross all these different systems. And you know, right now, for something like the time keeping system, it's like, you know-- project engineer defines a drawing. And then he has to go and submit a help desk ticket to one of us to enter a charge code, you know, somebody in IT. And so there's some really inefficient things that we still want to get cleaned up.

Many things between fileshares-- all kinds of simple things like this, so-- So wouldn't it be nice then, if all you had to do to not only integrate systems but speed up the ability to connect many different systems that are part of your process or part of your business, if you use a point and click and make that happen, right? So I was talking to Martin-- well, I mean, actually-- let me back up and tell you a story.

So I was mentioning Replicon. And so last year, my boss who's the VP of IT at Greenpoint-- his name's Mark-- I was offsite at the time. He called me up, and he said, hey, I just got an invoice for this development environment for Replicon. I don't want to pay it. Like, do you need that?

And I said, yeah, I need that. And he said, well, what are we doing with it? Well, we haven't done anything with that yet, sorry. And he said, well then, why do you need it? I said, well, we have plans to integrate you know Fusion Lifecycle and Replicon, but we just-- we have so many things to do and everything's so complex. And we just haven't gotten around to doing this integration project yet.

He said, well, you know it's like \$6,000 a year. I said, no, I actually didn't know that, but, yeah sorry. He said, well, look man, and I'm going to go ahead and pay this. You have one year to get this thing integrated. I don't want to have this thing next year. I said OK. And still haven't done it-- that was back in March.

But the point is that if it were literally about pointing and clicking, right, about essentially setting some fields over here, setting some fields over there, in order to make this integration happen, it would have been done long, long ago. But I've got 25 projects like this, right? So it's not just OK, yeah, I could probably sit down for a week and get this Replicon thing to happen in middleware and other things. But when you've got, you know, many, many, many different requirements like this, then some of them fall through the cracks. But this is \$6,000 falling through the cracks, every year.

So anyways, I was talking to Martin and you know, telling him about this problem. He was like, you know, he basically said this to me-- like you can't really do that. And then, he started

thinking, well, maybe you can. All right, I'm going to turn it over to Martin for a little bit. He's going to talk a little bit about Evented Web.

MARTIN

GASEVSKI:

Can you guys hear me? Yeah, Michael and I spoke at length about the integration. I think Greenpoint is probably the first company that started integrating PLM with. And they actually integrated PLM with Vault-- one of the very first ones. So we went through the challenges of integration very early on when we had-- when we offered Fusion Lifecycle on the market. And then, there was a challenging integration in a sense that we tried it before middleware, right?

So we looked at, how can we provide Fusion Lifecycle a good integration strategy that can be repeatable? And we found Jitterbit at that time. And we partnered with Jitterbit. And with the ease of their tool and the ease of their middleware integration solution, now we're able to repeat that business and integrate with on prem social, mobile, front and back office-- front office, back office tools.

And this is a great partnership. This works well. There's needs for these types of integrations-- complete heavyweight integration as that is between mission critical systems where data integrity needs to happen on one side, where the systems need to ensure the data is transferred all the way to the other side. And there's a closed loop exchange. But this is all great.

When we talked to the Greenpoint, they're like well, we challenge you to actually make it even more simpler. This is good. They've built IT on staff. And they've picked up Jitterbit on their own and started integrating.

They used consulting services at the beginning, right? And then, they got trained on it. And they took matters in their own hands. And now, they are actually running the show on their own. They're in full control of implementing their requirements. But the challenge was still there.

So it was kind of in the back of my mind. How do we-- how do we make this even simpler? How do we make it simpler, so that Michael when he talks to his boss about Replicon, he will say, well, let me show you a proof of concept that we can integrate it, so we can keep it another year?

So we looked at the systems, and we found different customers that say, I have-- when I met with a problem to integrate Fusion Lifecycle or any other system with, they're met with three

options. They buy IT on staff. They get consulting services. They deal with a problem. Basically, they say this is too complex too costly. I'm not going to integrate yet. I will hire humans, and then try to sync manually. Or they will get consulting services in and do something custom.

But that's not a story we wanted to do. We wanted to provide a story where knowing that Fusion Lifecycle-- the PLM system-- is not the only system in a given enterprise. It always sits as a system side by side next to CRM, next to an ERP, next to a PDM, whether it's Autodesk solutions or our comparative solutions.

We knew that somebody getting Fusion Lifecycle will eventually-- it's inevitable-- to have a need to integrate those systems. If they want to streamline their business processes, someone needs to start thinking about providing real time business integration use cases, ETL types of use cases, where data from a legacy system needs to go to Fusion Lifecycle, as well as mash up use cases where you're not necessarily transferring data between the Lifecycle and an ERP, but you're trying to get a view of the data as it's sitting in another system.

So all of the three use cases are provided by Jitterbit today. And it's a happy partnership. Can you go to the next slide? So the challenge is how do we work with all other products that Michael and his team would come across in the future, like Replicon, like any of these day to day products that you guys see that fulfill somebody's job?

Somebody has a private box, somebody uses [? Munchy, ?] GitHub, JIRA, Slack, right? There's a myriad of applications that are in use today, that are not integratable or are they? And the question came up, how can we integrate with all of these things in Fusion Lifecycle and better the day of those engineers, of those procurement guys, of all of the conversation that happens between a designer and a manufacturer? Next slide.

So we want it to integrate with everything. Actually, how many non Autodeskers are in this room? Can I see a raise of hands? OK, very few. Well, thank you Autodesk for your support.

So you must have seen this, guys? Or, a few of the Autodesk guys have seen this. What we wanted to provide is ability to do a non-programmatic way of lightweight integrations. Not the heavyweight builders that provide complex, conditional logic-- that say when I do something in Fusion Lifecycle, I need to go check out a file, extract building materials from Vault, transform the building materials and push it back to PLM, then grab that, put it through ERP-- none of those use cases.

We wanted to find something simple that will say, when I have data supplied in PLM, I want you to reach to Salesforce and update my vendor information-- a simple, clean, happy path use case. When something in system A, then something in system B. So that was the premise behind this.

Now, Evented Web-- I'd like to introduce the concept of Evented Web. It's not a new concept. I mean, it's a term that we're using for defining this solution. However, it's a very old technology. We're just trying to modernize it and put it into our cloud. We wanted to make it available and built for the clouds.

And we believe a modern cloud PDM PLM system such as Fusion Lifecycle requires to have something like this to do quick prototyping, to do quick value propositions to your upper management to say, hey, I can integrate with ADP. Hey, I can integrate with work very, very easily. Then, you can get some budget. And you can actually start funding that integration.

Moreover, it's not just for prototyping. You can start a lightweight integration and as your business needs grow, and you want to on board different departments into Fusion Lifecycle, you can start onboarding and make things make this thing more complex and more complex and more complex. Yet throughout this whole time, you're using an event driven approach.

Michael uses real time data. He needs real time data. He cannot afford to do polling of Fusion Lifecycle from Jitterbit, because even any polling to a PLM system or any other system, it actually creates resource constraints on that system. That user keeps polling and polling and polling, and actually, his users are struggling from the UI.

So we wanted to do a publisher subscriber model. And we applied that model into Fusion Lifecycle in the Evented Web. The concept is pretty clear. There's a trigger in one end. And then, when the trigger happens an action happens in another system. And the Evented Web is consisted of two components. There's a UI aspect to it. And then, there's an API aspect to it.

How many of you are technically savvy or have touched API and are comfortable doing API integrations? Perfect. Oh, many of-- most of you actually. So the goal was to say, we want to appease both personas-- the power user, the one that actually is not intimidated by looking at a rest the API and payloads and integrating and transforming, as well as the people who want to say, I can try to integrate with Box to Dropbox really quickly without any programming skill set whatsoever. And then, I may get further funding to hire somebody that has technical

expertise to extend that.

So trick is-- there's a trigger. There's an action. We wanted to provide bidirectional support all managed by the Fusion Lifecycle area so that you can trigger from any other application-- cloud, on premise, social, mobile-- and the action will happen in PLM.

An example-- one app that I mentioned the vendor. Now, when I updated vendor information in Salesforce, I would like you to sync up real time in event driven fashion the information about the supplier. Maybe the D-U-N-S number changed, maybe the address changed, maybe the name changed, right? So the information is constantly in sync.

The user interface aspect of this Evented Web is powered by Jitterbit. So we wanted to further enhance and harden the relationship that we have with Jitterbit. And in addition to providing a middleware, they will provide us the user interface that powers the applications. Next slide. OK, all right, perfect.

MICHAEL PARES: Thanks, Martin.

Just making sure-- I've got people messaging me. I'm making sure it didn't pop up there. So like Martin said, you know, he's talking about the different approaches. And you know, I wanted to just quickly talk about the three approaches to integration. And you know, certainly, we've used them at Greenpoint, you know, because Evented Web is new. We're not using it in production yet, but I've got it ready to go. And there is certainly the burning desire to use it, so.

But just sort of the-- I want to compare these three approaches for you. The first one-- scripting and custom API use as the sole way to integrate. Next is middleware, and then lastly, Evented Web. But with scripting/custom API, you know, for me and the way I view it-- doing integration just with scripting and API-- it ranges from pretty much moderately difficult to impossible. There are some things that we just won't be able to do with scripting and API alone.

Some of that may be due to timeouts. Timeouts-- 900 milliseconds I think is the scripting time for actions, right? 9,000 milliseconds? Well still, it's not long enough. I mean, if you want to do a scripting approach only-- it's certainly long enough for performance. You don't want your script running longer than that, because things will break, so but you're subject to that. When you write scripts, when you're doing integration, you have to think about that. It's, you know-- there's a limit to what-- how it can form.

And the other thing is you really need a developer. You need somebody who can write code and understands what they're doing with integration to achieve this.

For middleware, it's a little bit more simple-- takes a lot of the work out of it that you would otherwise do through scripting. So there's some plug and play aspects to middleware, which is great. It can also range to very complex. I'll show you a couple examples.

And but you still need an expert, or you need some expertise. You need somebody probably to show you around. At least, I did when I first started out with Jitterbit. I didn't really understand what I was looking at at first. It takes a little while to get it. Once you do-- once you pick it up, it's great. But you need to invest that time.

But lastly, Evented Web-- through the UI ideally-- point and click. And you know, who do you need for that? Person who can point and click, right? It can't get simpler. There's no simpler way to integrate two systems then doing that. Yeah, absolutely.

AUDIENCE: [INAUDIBLE]

MICHAEL PARES: Yeah, right exactly.

So I want to give you-- just to illustrate that-- I want to give you some simple examples of each of those approaches. I'm going to show you essentially how to do the same thing multiple ways, OK? So I want to show you how to send messages to Slack through scripting, through middleware solely, and then also through Evented Web.

So let's look at scripting and sending a message to Slack. What does it take? Well, first you can go to Slack. You establish incoming Webhooks over there. That's pretty easy. Then, you need to create an action in a library script. You need to have a function that you can call that takes the-- you know, essentially handles the HTTP call going to Slack incoming Webhooks.

And you need an ActionScript to call your function that provides it the message. This is the example that I came up with. Yeah, you could put it on an ActionScript, but that's not very reusable.

So I'm not going to encourage that kind of thing here. I do have a software development background. I like to see things be a little bit clean or reusable.

So this approach works. You could tie this to a workflow transition. And you could tie this to,

you know, creation behavior, on demand, anything like that. But this is just a simple example. This is just sending a message to Slack-- pretty straightforward, right? The things that you might want to integrate can be a lot more complex.

Middleware approach-- for us, it's about first having a way to trigger your middleware to operate. So I'm going to show you what we do to send a message to Slack using Jitterbit-- just the middleware. You're creating an endpoint in your Harmony API, which is just establishing URL where you can send the request.

And then, you connect that up to an operation in your Jitterbit project, right? What is your-- I said that-- basically, that that gives you the freedom to operate in any way that you like. And I talked about sending a message to Slack. This is that, you know, in actually two ways.

And so you can also get a lot more complex with your middleware, right? That's our Vault integration. I wouldn't necessarily want to call that-- yeah, that's not too simple-- simpler than trying to write all that in code, and in ActionScripts, and things in Fusion Lifecycle. But that took a while. But yeah, this is the operation to send a message to Slack. I'm just grabbing a source, transforming it into a very simple Slack JSON format, sending it off through that post.

So it requires minimal code-- minimal code, which is good-- you know, simpler than scripting. But what do we really want? I mean, if something is that easy to hook up, or you know, I don't want to have to hire a developer or hire a middleware expert or anything just to send a message to Slack. It should be like pretty easy right? Straight forward.

So that's where the Evented Web comes in. This is how you do it with Evented Web. I go over to Slack. I create a token. Copy and paste that token there, and hit Save. And I can attach that to creation, workflow transition. And any time something happens, then it's triggered in Fusion Lifecycle, and Slack gets a message. There is no simpler way than that to integrate two systems.

So that's good. And you know, Autodesk will have-- and Jitterbit will have-- an ever-growing inventory of systems that you can integrate with. But of course, it's not going to be complete at the moment that it's first offered.

And you know, you're going to have systems in your business that may not be on the list, right? But you want it. But you still want to integrate. You're still doing something fairly straightforward. You want to reduce the risk of timeouts. You don't want to go through a lot of

complex middleware, but you might have some ability to do some middleware work. That's good.

You know, so I came up with this idea of sort of, what if you don't necessarily have the point and click option, right, available? Then, you can still leverage Evented Web through the API using a sort of a simple formula. And so I want to show you guys. After, you know, I spent a lot of time working with Martin, working with Jitterbit, on sort of developing the instructions, the handout, and everything.

I don't know if-- has anybody downloaded the class materials or read them? No? Good, it's a lot of pages.

But I put in there really step-by-step instructions-- very simple-- of all the different things that you can do with this. So but I wanted to break it down for you in here. What's the formula? What's the easy way, right?

And it really comes down to three things-- this acronym ART. The ART of Evented Web-- Authenticate, Register, Transform-- you authenticate with Forge, right, forge.autodesk.com or developer.autodesk.com. You register, which is to actually attach your Webhook to something in Fusion Lifecycle, and transform using middleware.

When your event is triggered, what happens with Evented Web is it shoots out a payload-- represents the item that triggered the event. And so you need to take that, transform it, and then, send the message or conduct integration with whatever you want. So that's good. That exposes it-- that opens it up. Oh, I don't have to just go, you know, only do things with Slack or only do things with-- you can tie this to anything that you like, any system.

So for authentication, like I said, the steps to authenticate using Forge-- create an app on developer.autodesk.com or forge.autodesk.com. Request a token using-- and get that app client ID and Secret. And then, enter the-- sorry-- yeah, basically, authenticate. Like, put in your username and password to get your token. Right? That's with Forge.

And I'll show you guys. I've actually-- in the demo I'm going to walk through all these steps again and actually do it for you.

Register-- we're going to take that token that we got previously, use it in our header. We're going to POST, send a POST message to the Evented Web API. You can do-- you can attach to a create item event or workflow transition. You just have to specify a callback URL, which is

where you want Fusion Lifecycle to send the payload when your event's triggered. Your name for Fusion Lifecycle, and the workspace ID that you want to attach to or listen to.

And specifically, if you're doing workflow transition-- if you're attaching to workflow transition, then you need the workflow transition ID filtered to that. You could not provide that and register a Webhook on every single transition in your workflow. I don't think that's-- I can't imagine wanting to do that. That's going to create a lot of noise. I recommend you use filters anywhere and everywhere possible. Maybe you have a different use case, but you you'd probably say the same thing. You probably want to use a filter.

AUDIENCE: [INAUDIBLE]

MICHAEL PARES: There's-- yeah there's basically another object in your JSON payload in the body of your request I can show you for filter.

And then, transform-- like I said, I use Jitterbit middleware to transform. So when Fusion Lifecycle triggers an event, I'm going to receive the payload that it's sending out at my callback URL, which is what I had set up previously with the Harmony API-- that URL. I'm going to transform that payload with my middleware in a simple way.

The point I guess to make here is, I'm always getting sort of the same type of output, right? So I'm getting to do this transformation in a very consistent way, which means I can trigger all kinds of different things to happen off of a single Webhook event firing, right? And then, once I transform it, send that request out to whatever other systems I like. You know, that could be anything.

So I'm going to show you how I send messages to Slack through the Evented Web UI-- through just configuring the UI-- how I configure it, how I send the message to Slack. And I'm going to show you how that also can be done through middleware-- sending the payload to Jitterbit Harmony, and then transforming it, and sending something to Slack again.

So the first thing I want to show you guys is just the UI. And what does that experience look like, right? Because the ideal thing is if it's truly point and click-- and you know, I'm serious about it-- I should be able to do-- maybe I should stand up.

I should be able to do at least this part of the demo entirely with a mouse-- keep my hand over here somewhere off the keyboard, right? So I'm going to try and attempt that in front of you

guys. I haven't done it. I haven't practiced this, OK? I'm just going to do it, because I have confidence that it works.

So you know, what you'll notice is when this is turned on in your environment, then you have an integrations menu item here. And you have all these different options.

Here's Slack. And let's say I want to do this one. I create a Slack channel. No, that's not what I want to do. I'll go back the other way-- here. Create Slack message from new PLM customer. That's actually going to be change order. So and I don't have anywhere I can copy and paste the word change order, do I? All right, well, it will be mislabel.

But what I said I had to do was go and create a Slack token, right? And I should have favorited that. All right, so I'm going to go ahead and-- I have to use a keyboard, I am sorry, just to get my Slack token-- just to search for it. All right, so I'm cheating a little bit.

So I'm going to go over here to Slack. It's very easy to issue a token, just reissue a token. Right here I can click on that. I can select all this guy. I can copy it, come back over to Fusion Lifecycle right here, paste it.

For my Slack channel, I want this one. Can I copy this? Oh, come on. I don't want to type. I really don't want to type. Yeah OK, I guess I'll have to type. That's a bummer. I was confident that I could-- but then you know, technically I'm still clicking, right? Keyboard clicks are clicks, right? Save and finish.

All right so that's it. I've registered a Webhook there. So now if I go ahead here, I'm going to create a new change order, which is where this is attached. And just fill out the change order, hit save.

You're going see a bunch of things happening. But this right here that says Slack API tester-- this one came from my UI-- the configuration that I did in UI. Because I went ahead and bothered to do the demo in front of you guys and actually hit Save, you can see there's actually two messages that got sent, because I have one that I had previously registered in the same place. But this one was sent from there.

And then what I'm going to show you next is that this one came from a middleware approach, and how we achieve that. Does anybody-- how many of you guys actually have Jitterbit-- access to Jitterbit, use of Jitterbit or Fusion Lifecycle? A lot of you, OK. So if you already have

middleware, like Jitterbit, something like that, you know, this is for you. And if you don't have it, maybe you know this might be interesting to you if you're thinking about it.

But not too difficult to do-- so first thing that we're going to do in order to achieve this, I went over to developer.autodesk.com and logged into Forge. I create an app on Forge. Give my app a name, a description, a call back URL.

I'm using Postman. So I can put that there. This is whatever web site you would go to after. I thought it was valid. It wasn't valid? Oh, thank you.

I want to do that. And then, I can provide client ID and client secret right here that I get to use in my thing. I'm already got some configured here using my other apps. So I'm just going to use them.

You can see all the different types of calls that you can make for the Evented Web API. And I've got examples built out for every single one of these in the handout, so I'm not going to go into all of them. Right now, we're just going to do this create item Webhook right here.

Actually, maybe what I'll do-- oh, I have to speed up? We've got like 40 minutes left. You guys don't want to keep talking about this for 40 minutes? No, all right, so I'll speed up a little bit.

Since I already do have one registered, maybe I won't actually do the post. But I'll just show you that, you know, what we're talking about. So right here, I'm going to get new access token. I've got this stuff set up in here already. I'm going to request my token, right? This is developer API Autodesk.com authenticate, my client ID and secret that I just generated.

Like I said, you log in with your credentials, doing three legged auth, get my token, which is provided to me right here. Use token with this set to header actually puts that right here in the header for me, so I don't have to do anything, you know, crazy. I set up these other pieces of the header, but this is just to get a very simple-- hit send. This gets me a list of all the registered Webhooks on my tenant. Post-- very similar, same headers-- make sure you use the right token.

And then, the body for create item-- super simple-- here's my callback URL. This is my Harmony Endpoint that I had set up. Basically, if I send anything to this, then it triggers my operation in Jitterbit. And this is the workspace that I want to use.

And just so I could show you-- here is that filter-- what the filter would look like if you're doing

workflow transition. Again, all this stuff's covered in the handout. You guys can go and dig into it. So I send-- I use this post. Hit send. Register this Webhook to the change order workspace, and that event fired by Jitterbit orchestration.

AUDIENCE: [INAUDIBLE]

MICHAEL PARES: What's that?

AUDIENCE: How complex can you go?

MICHAEL PARES: There is a whole reference on GitHub help for it. Martin, you want to?

MARTIN [INAUDIBLE]

GASEVSKI:

MICHAEL PARES: So like I did Sean, when I showed you the Harmony API. It was configured to call this operation-- main incoming Webhook. In here is just a script, and all it does is run the next operation-- super simple. And it just says, basically, send global variable to file. The Harmony API has some predefined global variables, so for the incoming payload, you can access the headers, the parameters, the body of the incoming request payload.

So that's how I get access to the JSON body that came in, right? And when I just take that variable, it's contents, and write it out to a text file, because that's the easiest thing for me to do. You can probably do this a bunch of different ways. But that's the easiest thing for me to do. And then, grab that text file and transform it-- real simple-- with a little filter here that says, I just want the number field and the value for the number, which is that change order whatever 37 or whatever it is.

And then, here I'm just basically composing a little thing with the title and the value. And my JSON for-- let's see-- and then I send that to a temporary storage. But again, this could be sort any way you want to do it.

And then from there I'm triggering three different integrations. I'm sending a message to Slack, which is one we talked about. So I'm grabbing my little message from temporary storage and putting it into a Slack JSON format, which is literally just like an open bracket text field, and then, whatever you want to put in the text, and then close. And it's super super simple-- and then making that call.

Over here, I've got a little story to share-- and I will go fast-- about Microsoft Teams. And then, here is the Replicon one that I'm actually still working on. I've got it to work before, but right now it's got a little authentication--

So like I said, that one that you just saw in that-- come back-- that was this message that I configured. That just sends the number. And I also sent one here to Microsoft Teams. Any questions so far? No? All right, I'm going to try and wrap up here for you guys.

So you can see basically using that approach, you could pretty much do whatever you want. There's no scripting at all on the Fusion Lifecycle side in order to achieve that. That's important, because that means it's not going to time out, right? So you can send all the messages wherever you need to, including the things you can configure through the UI. You can register a bunch of hooks, but I only had to register one to send to Jitterbit and still have another three integrations completed, right?

So again, you know, just going back-- what was sort of the old way of thinking is this. And the new way that we're trying to achieve-- real time data from eventing, real time reporting, faster decisions, automating those decisions. You know, the difference between sending something to a reporting database versus sending it to somewhere where something's going to actually action-- take place-- like, in JIRA or something through this process.

Well, let me get back. There's one last story about this- sort of as an example of how this has helped speed up decisions, speed up my ability to react, my company's ability to react. Something happened just this week that I wanted to share with you guys which is, on Monday we're sitting in the CAB meeting. And you know, I've been thinking about this.

My boss contacts me and says, hey I want to try Microsoft Teams, you know. We haven't rolled out Slack to everybody in our company yet. And he says, well, we now have access to the Teams. I think maybe we should roll this out instead of Slack to everybody, because you know it's a Microsoft product. And we want to use all this stuff. And you know, free for us essentially. It's basically like a Slack killer. It's the same type of messaging app.

And I said, OK, well if it's going to replace Slack, then it's got to be good at this kind of lightweight integration stuff. It's got to be able to connect. And I want to send messages to it and whatever. So I went and checked it out. And found, yeah, actually, it's pretty easy-- sort of the same thing.

I can configure incoming Webhooks and send messages to it. And I did that, connected up through Jitterbit in about 20 minutes-- and then, spent the rest of the week kind of testing this thing, making sure everything's working, sending these messages, both to Slack and Microsoft Teams. It's all been working well.

And then yesterday, he messages me over Microsoft Teams. And he says, man, you've like Webhooked the crap out of this general, you know, channel on Microsoft Teams. And can you stop it? Like, why aren't you using the AU 2016 channel like you were doing over on Slack?

And I said, well, that's because you know on Monday when you first turned Microsoft Teams on, you only had a general channel. You hadn't created-- and you had these channels. And there wasn't an AU 2016 channel. He was like, well, oh Yeah. Ah, can you fix it? And it took me like two minutes to go back and reconfigure and change that integration.

So really the point is, reducing the cost to integrate, making it simple, and making it through the UI, or even through the Evented Web. It's not just setting things up for the first time, but all of the types of changes that happen in your business, much, much easier, less costly, faster-- to be able to react to those types of changes, and keep all the information flowing, keep the integrations happening. That was sort of the point.

And then, just a little bit about the process of getting here-- you know, I've had the opportunity to work with a lot of this, you know, preview or pre-release software with Martin, with Doug, and Jitterbit. And so I spend a lot of time, you know, I guess offering my opinions about documentation about things like that.

Red isn't bad necessarily. Like for me, I really like things to be in plain simple language. I really like, you know, to be able to understand from the documentation, you know, things like front to back. So I mean, it's a good opportunity for me to be able to, I guess, affect the outcome, to be able to work with them closely, explain all of the things that I really didn't understand the first one, two, or three times that I was going around trying to get these things to work.

And I've not only given the feedback to them, but I'm trying to provide it to you guys in the handout and in this session. That you know, you can start leveraging these things if you've sort of follow the right guides. There are some things that you can't do exactly for you.

So with that, are any questions before we move on?

MARTIN

Go back. Back. Any questions for this until so far?

GASEVSKI:

MICHAEL PARES: So I'm going to let Doug--

DOUG MCLEAN: Clicker and a microphone-- it's like two hands, two things at once. Thank you, Mike, for sharing that.

I think I wanted to just put things in context about the direction we are. We've had a relationship with Autodesk since 2012. It's been an exceptional relationship. A lot of-- it's helping us steer our direction. Go back. Go back.

So I wanted to start with this slide. And this was out of last year's Gartner integration platform as a service magic quadrant. I think the key message here was that there's a consolidation going on the market. And I highlighted that bottom section which was the key area. So what Autodesk has coined as the Evented Web is under a category of what Gartner is posing as citizen integrator.

So companies don't just want to buy-- you know traditionally, you had integration. You either coded it, or we sold you a tool, and you went out and built your integration. And so the market is just trending to where people again, they want to buy just the process. They want to buy these simple integrations.

And so this has become a strategic initiative for Jitterbit. We will be announcing-- we will be publicly announcing in the next three weeks citizen integrator and the availability of these recipes-- these pre-built processes-- will be part of our product. So anybody that's an existing Jitterbit customer-- that's bought through Autodesk or bought from us directly-- you will have access to start having these catalog of pre-built recipes. And the goal is to be building more of those.

I think this is relevant for both partners-- whether you're Autodesk, whether you're a system integrator, and you as a company, you as an individual, and user companies, have the ability to go build these repeatable processes. I mean, just for general context, those that you may or may not where kind of we fit-- I mean typically, you know, integration has fallen into one of these kind of four buckets.

Bottom left, you've got really the ability to go code yourself, right? And you can go build it. The challenge is, you typically are running into issues of speed. And the key thing is the

functionality. How do you maintain? And how do you manage it? There's certainly solution connectors. And somewhat, recipes fall into this category, right? Meaning that, I can deliver something to you very quickly, but it's a very fixed package of integration.

The old on-premise middleware solutions, again, typically very heavy, very developer focused kind of solutions. And so in general on our platform, the beauty here is we can kind of handle that traditional custom bespoke kind of integration. But we're certainly seeing this emphasis of moving more into these packaged solutions. And so that's where this whole concept with us of citizen integration is coming.

Finally, for those that may or may not know-- I mean that the areas that we see-- that we support-- I mean, you've got the traditional on the left there-- your traditional data integration, you're ETL, batch transform, extract, transform, and load. And then, the other points I would point out would be really this API integration, where you can publish real time live APIs. A lot of it's being used in the citizen integrate-- in the Evented Web process. I don't know why that automatically advances.

And the third key pieces are really that process integration. How can you manage and control and maintain the business logic inside of the Integration process? So whether it's dealing with, again, cloud to cloud, ground to ground-- with Evented Web, we're dealing specifically with cloud to cloud integrations. So there's being run on our multi-tenant cloud.

The final piece I'll call out really is around more for customers or partners. If you're in a business where you're providing a business services or you're a system integrator or you're an ISV like Autodesk, you do have the ability to be able to implement and support multiple customers across. So again, that's basically our processes or the things that we cover.

Thank you very much for attending in the time, but certainly go back to Mark. Why is that on a timer there? I'm not sure. But I'll turn it back over.

But thanks so much for having us. And certainly, if there's anything I can provide-- I'm the Alliances Director for Autodesk, so you work with anybody that's an Autodesk customer, partner, or any of the other multiple divisions in Autodesk. So thanks for having us.

MARTIN
GASEVSKI:

Thank you Doug. So Doug just announced to you that if you are an existing Jitterbit customer, you will get access to a portal where there will be multiple applications working with them as a partner. And then, Michael mentioned and walked you through a scenario where Fusion

Lifecycle has pieces of that portal embedded within Fusion Lifecycle. So those applications that you see where Michael is going from Slack or NetSuite or HipChat or GitHub and JIRA that you guys saw, those will be the very same packages that appear on the portal.

To extend to Michael's story is that Fusion Lifecycle, Fusion Team, or any of the Fusion families could become a member of the Jitterbit's Harmony citizen integrated portal. We're not there yet. This is what we're working with them actively. Right now, we're working on API availability to build custom Evented Web solutions, even to standards functionality, also some limited prepackaged applications with recipes as Michael showed you with a three click use case for Slack.

I wanted to give you some insight, that we started building an infrastructure behind this, so I wanted to give you just a quick overview of what's under the hood for the Fusion Lifecycles Evented Web Webhooks. So this is our big box over here on the bottom right corner is the power that drives Evented Web. And it's built independently of any of our Autodesk cloud systems over here. So this will be Fusion Lifecycle, Fusion Team, BIM 360, whichever the application that supports Evented Web or an application that has ability to register hooks to would be able to leverage the right side and the service of the Webhooks infrastructure.

If the components right now-- this is all in motion. But components are [? cassandra, ?] so we are expecting massive scalability. So we can withstand and support events of a scale of Facebook's or Twitter's. The event queue is right now Amazon's SQS system, so as events get queued up by n number of applications, we can actually queue those up and support massive scales of events.

Another custom microservice has the intelligence to listen to events that coming from any of these supported products, and will say, OK, this is an event from Fusion Lifecycle. I will go, and call back the registered URL, and it will reach the targeted application. So a similar thing will be happening on the other end, when we want the event from Box into PLM, it just will be listening in events that Box would call up into our event queue. So this is a kind of a generic way of doing things.

What I'd like to also announce is that the-- how many of you have heard of the Forge platform, the product innovation platform-- future of making things, right? So the Forge platform is our answer to-- is our execution answer to a story of how do we implement the future of making things? And when you look at-- can you go to the next slide-- When you look at the products of

Autodesk's that are in the Fusion family, we really want to make a cohesive, integrated experience between the products.

So as you see Fusion Team, Configure, which was our configure [INAUDIBLE] position, Fusion 360, Lifecycle, Connect-- our IT strategy-- these products work on their own. They're independent solutions today. And we're working hard to connect them. What connects these products is the data that's in the middle.

The customers that are attending right now, you do want to manage your products. You want to ideate over the products, socialize over the products, review them, collaborate over them, maintain the data, maintain in the lifecycle of those products, even through to obsolescence. What we are working on is unifying this experience between the products so that you-- at the end of the day, the data will be federated for you.

How do we do that? Well, the Forge ecosystem provides you the range of the services. And you've probably seen this slide throughout these past three, four days. The Forge is a-- the Forge is a brand. It's an Autodesk brand. Underneath that brand, you can think of this entire spectrum of services as Autodesk as a service. These are the services that we will provide for you to access, just as well as our Fusion family access them and federates the data across of these services.

Now, what we're working on in addition to Fusion Lifecycles Evented Web, we're working on a model where an eventing service will be available for all of Autodesk products, not just the Fusion family, but as well as for many of these Autodesk services where it makes sense. For example, the data management service over here in the top right corner has a need for eventing.

When you work with Fusion Team or BIM 360 Docs-- I've attended some classes, and I've seen, well, how do I tap into BIM 360 Docs? And there's some interesting events I want to listen on and react into Fusion Team, or reacting to Fusion 360, or react to Box, right? Or send me a text message when a contractor or subcontractor submitted some quotes that I need to review. These are the easy use cases they would like to solve.

With that said, I would like to create an event when a file has been modified into this folder or a file has been renamed or changed with a filter that says, when that change has happened by such and such person or such and such group, then fire an event. Where does the event go? It could go to a custom middleware solution. It could go to an existing third party established

application supported by Jitterbit's portal. Or it could go to Fusion Lifecycle using the Fusion Lifecycle embedded.

The key point I'm making here is that Autodesk has invested in Evented Web as in a sense of Forge Webhooks to provide glue between products and services for functionality that we do not already have yet. So we will continue to be building this cohesive integrated solutions, where you can go and out of the box receive the goodies between Team and Lifecycle and Connect and Configure. Yet the Evented Web and the Webhooks of Autodesk will provide you that the gamut where you can play and customize the events and connect things that we have not yet connected.

So over time, we will be understanding those patterns that you guys are using in terms of connecting applications and services that we have not come up yet, or that are not on our road map. And then, we can materialize those and put them into the product as we need to.

Next slide. So Michael is a member of our CAB-- Customer Advisory Board. He's privileged to look under the hood of how we develop things, how we manage our road map, how do we strategize, how do we prioritize things that are coming on the next three months, six months, a year. But also Michael gets to see all the delicate aspects of our development. And he gets to see the good parts as well as the bad parts.

The nice part with our partnership here is that we work together, and we give access to him to pre-release information, the pre-release tools, the pre-release functionalities. And even though this is not available on beta yet, we're planning to work with Michael and all the feedback that he has provided to us when it incorporates for us to say, are we ready for a beta yet?

So I'd like to mention that if you're not signed up for the beta of Fusion Lifecycle, please do. Because-- and monitor over there-- because in several weeks we will be announcing availability for a private beta available upon request. So we can reach to our support technicians, and we can turn on integration or Evented Web for you tenants.

As any beta, it's also a pre-release product, so the recipes that are over there coined and hardened may not make sense to you. But I do want to get feedback from you to say, I don't use Slack. I use HipChat. Can you add HipChat? I don't use-- let's say I don't use GitHub. I use Perforce for my source code management. Or I use GitHub to do my source code development over there.

I'm looking for feedback as you enter the beta portal. What applications are relevant for you? And exactly what use cases do you want to have out of the box for those applications?

So I'd say a bit about our beta capabilities. The triggers that we support are create item and perform workflow transition. So you will be able to register a hook on any workspace such that when an item is created in that workspace, PLM will listen internally-- will look internally to say, are there any registered hooks against this item that just got created? Yes, fire all those callbacks to Slack, HipChat, GitHub, whichever external applications want to be notified of such. PLM will make sure through the Webhooks service to notify those.

Michael showed you an example where he created custom single hook to PLM. And he decided to event out through Jitterbit. And within Jitterbit, he created a fork to notify three systems by himself. He could have done equally register hook for Slack, register separate hook for Microsoft Team-- Team office, Team-- Teams, and then register separate hook for some custom application, right? The flexibility is there for you to explore and tailor how you fit the needs for your business.

So the actions-- I would say this is a misnomer. We are working on a V3 API of ours. And action means an incoming message or an incoming rest call to Fusion Lifecycle. So if you use our solution, you will be registering a hook against Box or GitHub or Salesforce or NetSuites. And then, when something happens in NetSuite, the callback will be towards PLM. So our V1-- APIs are very capable that you would be able to stress that V1 API and make an action into PLM.

So with that I believe, are we at a rap or not?

MICHAEL PARES: Just about there.

MARTIN Any questions for this before we get to the next phase of this?

GASEVSKI:

MICHAEL PARES: If there's no questions, do you guys have ideas that you want to spin? And also, let them know where we're going to be after this.

MARTIN Yeah actually, I'd like to hear some questions, some comments, about this right now. If you

GASEVSKI: think of anything that's--

MICHAEL PARES: I have some more questions.

MARTIN Sure.

GASEVSKI:

MICHAEL PARES: I mean, so I want them to [? V8. ?] Like, I want to be able to listen to--

MARTIN Your microphone is off.

GASEVSKI:

MICHAEL PARES: --from outside systems, right? I want to be able to trigger something to happen inside of Fusion Lifecycle when something happens in JIRA, right? I want some bi-directional. I don't want it to always just be from Fusion Lifecycle going out.

MARTIN Makes sense. That use case has come up from him. We validate the use case with our other
GASEVSKI: CAB members, because there's a transparency there.

The use case usually comes with [INAUDIBLE] systems that build products that have mechanical, electrical, and software pieces. They have software functionality. For example, they manage their source code in GitHub, and they manage their development and testing of that source code and the quality in JIRA, in Confluence-- Atlassian, Confluence, JIRA, right?

So they want to say, I will create a test item in PLM, event out to create an equal test plan or something in JIRA, and create some stub in GitHub. But in JIRA, the product itself has its own workflow. And it supports its own Webhooks capabilities.

So what Michael wants us to use this same infrastructure to ask JIRA to say, when your workflow of your test plan and all of the churn that happens in JIRA [? lays ?] stay there, when the ticket gets closed, when the test is done, when it's completed successfully, event out to PLM and update information in PLM. So this is sharing information bi-directionally.

MICHAEL PARES: Would you guys find that valuable? Something-- I mean, not necessarily JIRA, but in other contexts, yeah?

AUDIENCE: Can we do that through the [INAUDIBLE] right now?

MARTIN You can, absolutely right. We talked about the three stacks. If you have a technically savvy
GASEVSKI: team, you would say, I can call the custom application to exactly develop the aspects for JIRA. I will call the customer aspect to receive the JIRA events, and I will do all my manipulation to event out to PLM. And I'll need to code.

The trick behind this is rapid prototyping. So if you want to do a proof of concept, you can actually do it with mouse clicks effectively. If you want to extend it further, you can keep the event registered in both places, and you can now enter under the hood and start extending on what's already there.

The beauty behind this is that you're not going to tear down your existing integration. You can start very simple. And then, you can add on functionalities. Sometimes you may need to say, I'll need to pause. I will consult to another system, do some transformation, and then, let the information follow through to the other end, right?

And as the event gets complex, you get into a situation where you say, well, I need some data integrity. I need to ensure that data reaches the other end. Now you can say, now middleware comes to place. [? Here ?] with middleware where it needs to do eventing messaging, repetitional stuff, conditional logic.

Even then, that's why we partnered with Jitterbit-- even then you can simply extract the simple event that you've started with. You can-- you will be able to extract the pack and import it into your own full blown Jitterbit middleware and continue building through middleware. So the model scales for us to start with custom API, meaning raw development, Evented Web lightweight integration all the way through to full middleware integration.

Did that answer it? I can't see you. I have to move over here. OK thank you. Anything else?

MICHAEL PARES: Yeah, I got another one. No.

MARTIN Go ahead.

GASEVSKI:

MICHAEL PARES: Oh, we can we can talk about it later, Martin.

So I mean, thank you guys so much for coming. It means a lot to me to have you guys here. It means a lot to me if you can give us some feedback. Please you know, if you want to influence me, to influence Martin, then please give me positive feedback. If you want to see us again next year, that's good.

And yeah, I'm just as interested in hearing from other customers as Martin and Jitterbit are. I want to hear what things that-- what kind of information we can share, what ways I can help

you. Please let me know if you have any questions about any of the materials or anything. I'd be happy to answer. And Martin, you're going to be the answer bar?

MARTIN

Yes, at 2:30, so right after this class, I'm in the answer bar until 4:00. So any questions,

GASEVSKI:

comments, if you want to know a bit more, if there's anything that comes up on your mind that you're not sure if it's Evented Web or not? Does it have to be sold with Evented Web? I'm over there. So thank you guys.

MICHAEL PARES: Thank you guys very much.

[APPLAUSE]