



.NET Rocks!
The Internet Audio Talk Show
for .NET Developers
With Carl Franklin **msdn**
and Richard Campbell
<http://www.dotnetrocks.com>

<http://www.dotnetrocks.com>



Carl Franklin and Richard Campbell interview experts to bring you insights into .NET technology and the state of software development. More than just a dry interview show, we have fun! Original Music! Prizes! Check out what you've been missing!



Text Transcript of Show #323
(Transcription services provided by [PWOP Productions](#))



Issam Elbaytam and Scott Wileke
March 11, 2008
Our Sponsors



<http://www.devexpress.com>



<http://www.code-magazine.com>



<http://www.telerik.com/>



Lawrence Ryan: The opinions and viewpoints expressed in .NET Rocks! are not necessarily those of its sponsors, or of Microsoft Corporation, its partners, or employees. .NET Rocks! is a production of Franklins.Net, which is solely responsible for its content. Franklins.Net - Training Developers to Work Smarter.

[Music]

Lawrence Ryan: Hey, Rock heads! Quit sliding your Gantt chart and listen up! It's time for another stellar episode of .NET Rocks! the Internet audio talk show for .NET developers, with Carl Franklin and Richard Campbell. This is Lawrence Ryan announcing show #323, with guests Issam Elbaytam and Scott Wileke, recorded live, Monday, March 10, 2008. .NET Rocks! is brought to you by Franklins.Net - Training Developers to Work Smarter and now offering SharePoint 2007 video training with Sahil Malik on DVD, dnrTV style, order your copy now at www.franklins.net. Support is also provided by Telerik, combining the best in Windows Forms and ASP.NET controls with first class customer service, online at www.telerik.com, and by CoDe Magazine, the leading independent magazine for .NET developers, online at www.code-magazine.com. And now, the man who got caught shaving his cubes, Carl Franklin.

Carl Franklin: Thanks very much. Welcome back to .NET Rocks! Carl Franklin and Richard Campbell here talking .NET for the next hour or so. Hey, Richard, what is up, man?

Richard Campbell: Not too much, bob. Having a good time, fixing hardware. I'm finally home for a few weeks, so I'm tearing apart machines.

Carl Franklin: I've been recording shows left and right, of course, and we did a Mondays and I edited that and that was fun.

Richard Campbell: Yeah. It's been five months since we did a Mondays.

Carl Franklin: Yup, but, you know, if you hadn't died, everything would have been okay.

Richard Campbell: Yeah, that kind of sucked. It was a surprise to me.

Carl Franklin: For those who know what Mondays is, it's a comedy podcast that we sometimes do. It started off weekly and we sort of took a break.

Richard Campbell: We did like eight of them last year.

Carl Franklin: It's not safe for work, however, but if you want to laugh on your headphones, definitely go to mondays.pwop.com and listen to that. All right, let's get right into Better-Know-A-Framework, Richard. Today's class isn't actually a class, it's a method.

Richard Campbell: Oh.

Carl Franklin: It's a method on the AppDomain object and we were talking about that.

Richard Campbell: Oh, yeah, yeah. You've had an AppDomain thing going for a while now.

Carl Franklin: Well, yeah, there are so many cool things in there and, you know, AppDomains became really important in the remoting space.

Richard Campbell: Right.

Carl Franklin: Because remoting is cross AppDomain communication basically, but I mean you still have the need to know what's going on within your context, right? So, there is a way to execute code in another AppDomain, from your AppDomain to another AppDomain and there's actually a sample and the method here is the DoCallBack Method.

Richard Campbell: Oh okay.

Carl Franklin: There's an example, just a few lines of code, just to get the current AppDomain just from AppDomain.CurrentDomain and then create a new AppDomain with CreateDomain and you give it a name and then you basically do a DoCallBack, right?

Richard Campbell: Cool.

Carl Franklin: And you pass the address of a subroutine and then that other domain calls your code in your domain, so it's pretty cool. It's just a neat thing that I found in the docs today.

Richard Campbell: What a very clean way to do a handoff between AppDomains.

Carl Franklin: Yeah and there's a lot to it though. I mean that sample looks simple, but there's a lot to AppDomain, so I would suggest reading up on it.

Richard Campbell: You have to do it right.

Carl Franklin: Right. There you go. So, Richard, you got an email for us?

Richard Campbell: I do indeed and this one's from Shawn McCarthy. It says, "Hey Richard and Carl. I had a little debate in my head about whose name to

put first. Being a fellow kiwi, I had to go with you, Richard. Are you two still going to do that show from your uncle's farm in Tauranga sometime?"

Carl Franklin: Oh, man. That would be so fun.

Richard Campbell: You like to, would you?

Carl Franklin: I want to do it from the front porch, man.

Richard Campbell: You would do it on the front porch with the goat out on the front yard there with the goat?

Carl Franklin: Yeah, I want to hear the ambiance.

Richard Campbell: Oh, the view from there is amazing, looking down on Tauranga. That's a whole other story. "I've been listening to the show for about a year and I really enjoy the edition of Better-Know-A-Framework section."

Carl Franklin: All right!

Richard Campbell: "Even the music."

Carl Franklin: Hey. You're okay then. I like this guy.

Richard Campbell: "In show #320, the John Bristowe interview, Carl mentioned the System.IO.Pipes namespace. You're right about pipes being a bit redundant in the world of TCP/IP for networking, but they are still really useful for local inter-process communication."

Carl Franklin: Yeah!

Richard Campbell: "Local named pipes live in the kernel and effectively become a shared area of memory between two processes."

Carl Franklin: Very cool.

Richard Campbell: "This is faster than going all the way down the road of packaging up all the data for TCP/IP transmission only to send it to a local host."

Carl Franklin: Yup.

Richard Campbell: It saves some serialization effort, right?

Carl Franklin: Yeah. It's a way to burrow through processes.

Richard Campbell: "This is a nice little performance tweak when you are running two major components of your application on the same machine, for example, a web server and business logic service. Of course, the best option nowadays is to use Windows Communication Foundation."

Carl Franklin: That's right.

Richard Campbell: "The way you connect is abstracted away from you in bindings which are stored in the config file. This way, you could easily switch from using named local pipes to either TCP/IP or web services when you purchase that second server."

Carl Franklin: Yup.

Richard Campbell: "Thanks and looking forward to the next show, Shawn McCarthy."

Carl Franklin: Shawn, you get a mug.

Richard Campbell: Ah, mug that man!

Carl Franklin: Mug that man!

Richard Campbell: And if you want a mug, send us an email at dotnetrocks@franklins.net.

Carl Franklin: Well, Richard, I've been looking forward to this show for a very long time. Our guests are Issam Elbaytam and Scott Wileke from Data Dynamics. Issam is Founder and Product Manager at Data Dynamics. He is a 20+ year veteran of software development. He started writing scheduling software in Clipper 87 for healthcare...

Richard Campbell: Love it.

Carl Franklin: Yup, you guys are going to have a lot to talk about -- and moved to VB3 during a brief consulting career. For the past 12 years, he has been helping developers and businesses make the most of their data to support decision makers. Scott Willeke is Director of Development at Data Dynamics. He has over 12 years of software development experience and has worked with the .NET Framework since early betas. He is currently leading multiple teams to bring next-generation business intelligence, data analysis and reporting components to .NET developers. Well, guys, we decided to have this show because, as our listeners know, you guys were our first sponsor and I can't say it enough, but I don't even know if we would be here today if it weren't for you guys. Is that a fair statement, Richard?

Richard Campbell: I think that is a fair statement. Data Dynamics were the first people that really believed in .NET Rocks! in those early days and let us



think, "Hey, we could just keep doing this, not just as a labor of love, but as a substantial part of our working week."

Carl Franklin: Yeah. We had a year to find a sponsor and if we didn't have one to help us keep going with the bandwidth and all that stuff, we literally would have been paying out of pocket and might have closed down. Anyway, the reason that we wanted to do this show is that I went to see you guys, Issam, I went to visit you at your facility in December to see what you guys have been working on because we'd seen you at TechEd and stuff, in your booths, and had a sneak peek about your new analysis stuff and it looked really interesting. So, I wanted to just get the big picture, so I went to visit you guys, a great company and a great group of people. The stuff that we're going to talk about today was just really fascinating to me, so I want to just welcome you to the show and let's dive right in.

Issam Elbaytam: Thank you, Carl and Richard. It's been really an honor to be right there with you guys from the beginning, you know, and I think the relationship has been a two-way street, really. I mean it's true we helped you, but we also got a lot of help, you know, from you guys being true believers in the product and not just basically, you know. I know that both of you have used ActiveReports and also helped us push it to many, many developers with your show.

Richard Campbell: I know the show is not specifically about ActiveReports, but just to call out to folks, I know they probably heard the ads, but the thing that both of us love about ActiveReports is actually reporting designed for developers to work with.

Carl Franklin: Right.

Richard Campbell: That was the thing that made it so compelling to use was it just was so natural to add reporting into our .NET apps where usually with other products, it was such a struggle.

Carl Franklin: We don't know what other products you're talking about, dude.

Richard Campbell: I wasn't going to name names.

Carl Franklin: The classic.

Richard Campbell: But I've been through them all, let me tell you. I've had a lot of pain. The scars are never going away.

Carl Franklin: It should be crystal clear who we're talking about here.

Richard Campbell: Geez.

Scott Wileke: Very subtle, Carl.

Richard Campbell: That's nice, Mr. Franklin. I like that one a lot, but really the funny part here is that Carl is the one who got to go to the office and see the cool new tools and I'm the data analysis guy.

Carl Franklin: Right, so you're really going to be in for a treat, Richard.

Richard Campbell: Yeah. I'm just totally flipped out to see what you guys have been up to.

Issam Elbaytam: Well, you're welcome here anytime you want, man. You know, we'd love to have you.

Carl Franklin: The thing that struck me when you were showing this analysis stuff is that this is really an application that belongs in a different market and it really belongs in the IT market, in the business intelligence market, and it's an application that I've seen similar apps that go for hundreds of thousands of dollars like it's a big deal for analysis, but yet you chose to wrap it up in a control and practically give it away to developers.

Issam Elbaytam: Exactly. It would help really to tell you a little bit about our strategy and our way of thinking really and how we got to this point. I mean bringing such powerful tools to developers has always been our goal from inception. We started about 12 years ago with -- I brought in some of what I knew or struggled with while doing consulting and development in those days and, really, I used to look at things like the simple pivot table, you know, in Excel. In those days like in 1994-1995, little people really -- it was not the hottest feature in Excel, but when I saw it, I saw it as the coolest thing ever, you know, but I could not use it. I could not use it in my VB 3 apps, I could not use it in anywhere really. So, I got together with my other partner, Ferhat, who you probably know and he was a hardcore coder, you know, a real C++ guy and I basically claimed idea to him, you know, that basically if I could use this pivot table in my application, that will be awesome because it would help not just -- if you notice a lot of applications, you just dump data into them. I mean most applications have one function, basically it's data entry. Accounting systems, you just put data in them. The only way for you to get information now has been basically, "Okay, so print me a report that does this," or "Print me something that does that."

Carl Franklin: Usually involving a little code or a little query or something.

Issam Elbaytam: Exactly, exactly, but really, the data that you're collecting throughout the years is not

really helping you make any decisions. It's not helping you really run your business. Basically, you just end of the month, you print your aging report, you print your P&L and these types of things, but you're not really using it day to day. So, all these people are collecting this valuable data and putting it into your computer systems, but not doing much with it. So, the pivot table to me was something that we could use to get more information out of data that is just sitting basically in this application. So, sure enough, Ferhat, he got on the job and he was able to do it like in, I don't know, it took him I think a couple of months. I did all the QA. We were just two guys, you know, with an idea and we did it and when we wrapped, it was DynamiCube. We called the product DynamiCube at that time. It was the first 32-bit .OCX for VB 4 in those days.

Carl Franklin: Oh really?

Issam Elbaytam: Yeah. So, we did not even start with a VBX. Most of the products at that time, they were VBX and they moved into .OCX.

Carl Franklin: Right.

Issam Elbaytam: We were actually the very first 32-bit, no 16-bit, .OCX control and it started with VB 4.

Carl Franklin: Wow. I remember those days.

Issam Elbaytam: Oh yeah. I remember when in 1996, we went to our first VSLive! We had to be at VSLive! and we went there. We did not even have a company name yet, so we get to the show floor and we came up with the name of the company right there, you know? "What shall we put on the sign?" then we decided, "Okay, well, we'll call it Data Dynamics." "Okay, sure, put it up there." That was the first of Data Dynamics, right there on the show floor of VSLive! in 1996. DynamiCube was a tough sell, man, because we were bringing something that no one really has ever considered from the developers like, "Okay, so, I know I need a grid and I know I need a report writer and I know I need these assured in my application," but no one considered having a two-dimensional grid that you use whose function is not to input data or to look at data in rows or tables, you know, but whose function is to analyze the data and make decisions based on it. So, it was really a tough sell.

Richard Campbell: You were too far ahead of the curve.

Issam Elbaytam: Exactly.

Richard Campbell: Because at that time, we were just trying to -- I was working in the analyst business

working with guys like MicroStrategy and those sorts of folks.

Issam Elbaytam: Exactly.

Richard Campbell: That were just starting to bring the concept of analytics to the enterprise and here you were selling it to the VB developer.

Issam Elbaytam: Absolutely, yeah.

Carl Franklin: Now, what's up with that?

Issam Elbaytam: Yeah, man, it was mind-boggling, you know, because you had MicroStrategy - you mentioned MicroStrategy and like Cognos and Holos in those days. They were bought by Seagate at that time.

Richard Campbell: Yeah, they eventually all got rolled up, but there was even more, right?

Issam Elbaytam: Oh yeah, oh yeah.

Richard Campbell: There was Brio and, oh my goodness, there were so many companies back then and most of them are gone now.

Issam Elbaytam: Exactly because the industry has consolidated, but we had a different idea. Instead of taking these things and make them IT applications that you sell for hundreds of thousands of dollars and you got to get a consulting team in and they install it, it's a mess. We thought, you know, why not bring it down to the person that is writing at the department level. He's got a simple database, maybe an Access database or a small SQL Server or really an ODBC. We even had like, gosh, I think Borland -- what was it called? BDE? The data engine? Yeah, we even supported that.

Richard Campbell: Right.

Issam Elbaytam: The whole thing ran in memory and it was so tight. I mean Ferhat wrote some of the routines and assembler and we went crazy with it, you know. We did not write it in MSD or ATL. There was nothing at that time. Ferhat had to code it right there against the OLE SDK and build the control basically...

Carl Franklin: Hardcore.

Issam Elbaytam: Yeah.

Scott Wileke: This is in the day when I think the system requirements for memory were 32 megabytes on these machines. So, it was difficult to fit in there.

Carl Franklin: Scott, were you around in those days?

Scott Wileke: Yeah, I was around in those days during tech support.

Carl Franklin: Okay.

Scott Wileke: I was able to just answering questions and trying to explain to people what this is and they've called up and say, "Well, we have Cognos installed. What's the difference between this product and yours?" That's a long difficult question to answer.

Carl Franklin: Yeah, especially because you guys are in different markets and you told me, Issam, when I was there in December that you just really like the developer market. It's what you know and you didn't want to swim with the sharks really.

Issam Elbaytam: Exactly and the organization that we're trying to create is not a huge organization, but more a place where we could have fun. We could produce products that we would want to use, you know?

Carl Franklin: Yeah.

Issam Elbaytam: Because I remember in my development days, I wanted something that I could use and actually build applications with. So, sure enough, you know, I got together with Ferhat and we put together a team. Scott was our employee #1. We hired him right out of high school.

Carl Franklin: Wow.

Issam Elbaytam: And basically, he's been with us, you know, grew from tech support. He could tell you a lot more about his history, but he's definitely one of those brilliant guys, one of the best developers that I know. I watched him, you know, just grow from tech support all the way to managing all of our teams and he's responsible for many of the products that you see today.

Carl Franklin: So, you talked about DynamiCube. Let's talk about Analysis. This is your new stuff.

Issam Elbaytam: Okay. So, basically, DynamiCube, we built it and it was an ActiveX control. It had quite a few firsts, actually. It was the first control that would run in Internet Explorer. If you remember, when Internet Explorer came out to battle with Java, they said, "Well, we can host ActiveX controls," and that was when the term ActiveX came out. So, we were one of the first controls to do that also, to live inside of Internet Explorer. When it came

to .NET, everything changed really. I mean what we noticed based from our experience with DynamiCube is that people see these numbers, but when you look at a lot of numbers that are sitting on the screen, many years of product analysis that may be category or break it down by region and all these things. It is not very easy to spot trends or problem areas. At the time, I read some research papers about a project at Stanford called Polaris. Basically, the idea was that you take any numeric values and you translate them into a graphical view, the graphical view will allow you to spot trends a lot easier. For example, to look at a series of let's say two years' worth or 8/4 of sales, you will see eight numbers. You have to actually focus in trying to see whether you had a growth or a downturn in your sales, but if the data was represented with a line chart, that would be a lot easier to spot immediately how the curve is going up or down. The same thing, you know, if you use the bar charts or some other, you know, depending on the type of data that you're looking at, you could represent it with a graphical view or textual view based on your need and how you want to explore the data.

Carl Franklin: And the thing we've got to stress here is that this doesn't take code. I mean this is literally, Richard, an application in a control, so you pick your database, you connect to it, you pick your stored procedures or your tables on either axis and then you pick your line type and your ranges and some other parameters and the data is just there.

Issam Elbaytam: Exactly. So, what it does basically, as you said, you point it to your data source and the data source could be pretty much anything that has an OLE DB data source or analysis services. It actually supports multidimensional data sources and relational data sources, so if you happen to have Analysis Services server that you've invested so much on, but you don't really have the frontend, how can you see this data? I mean the only way today to really look into analysis services data, these cubes that you've been storing a ton of information in is really using Excel pivot table, the enhanced one that is in 2007 or some of the just ProClarity stuff that they came up with, but either of those really requires so much investment both in money and time to understand and to make available to users. So, what we came up with basically is we built a control that you point to the data source and then you simply just start dragging and dropping to build this view and you could do it in either your Windows application as Win Form app or Web Form app. So, you could have a full AJAX frontend in a web browser or a Win Form application if you want it to be inside your, for example, accounting system or whatever it is that your application does and just let the user at it. You don't have to write reports anymore. They could just pivot around, drill down, filter, sort, create charts, do all kinds of things from this really amazing control.

Richard Campbell: Well, and the motivation here, I've been the guy and I'm sure a bunch of our listeners are the person right now that acts as the pivot cube. Your boss asks for a report, you whip it together with a Smart Query and a little bit of visualization, you hand it to him because "this is great. Now, can you change the following? I want you to break this down," and now you're acting as the cube.

Issam Elbaytam: Exactly.

Richard Campbell: You keep doing the report and I got so good at it, I was doing a new version every 20 minutes or so, which I realized was a mistake because it just encouraged him to do it more.

Scott Wileke: Now, that was something interesting though that with DynamiCube, like Issam mentioned earlier, it was really ahead of its time and especially it's definitely a niche market for developers who are way ahead of its time. It was difficult for us to get them to understand how valuable it was and later, when we added Active Reports, customers started to understand when they would say, "I want my users to be able to change what they're looking at and add different fields to their report and all this stuff." We allow them to do that and Active Reports as well with an end user designer, but sometimes you just mention, "Have you ever looked at DynamiCube?" A lot of times, they would look over at DynamiCube and realize that although it's not exactly a report writer in the traditional sense, but it's very valuable and they complement each other very well and it allows their users to actually quit asking the developer to make minor changes to a report and allow them to do that analysis themselves and at the end of the day, those programmers are heroes.

Richard Campbell: Right and you get to the real point here, because everybody thinks they want a report writer when actually what they want is an analytical tool. As soon as I create a report and a guy loves it and still wants changes, he's doing analysis.

Scott Wileke: Absolutely, yeah.

Issam Elbaytam: Absolutely, yeah. Basically, they want to reach their data, you know? This data that they have here, now, as a developer, I don't understand the business, so I don't know what it is to look for so I cannot create reports for him, so he's going to have to tell me what he wants in those reports, but what he wants is so random especially in a dynamic business environment. You cannot simply summarize that and have it fixed as a single view. That's why the idea of multiple views, a pivot table-like view, you know, the way they can grab the data, summarize it, and then I'm looking at a sum, I want to see an average, I want to drill down from a single cell

onto more detail. That is not something that you could do with a report. That's not something that you could do with a static view.

Richard Campbell: Right.

Issam Elbaytam: And for you to have a developer sitting there keep asking him, I mean it gets pretty expensive very quick, you know?

Richard Campbell: Yeah, it's not efficient use of time, but it also gets back to this idea that reports are what middle managers and administrative folks need to know what to do next where senior management, they don't know necessarily what they're looking for. They're looking for the exception. They're drilling around the data to find out what's wrong here and they don't know what the answer is. So, of course, the report never stays static because they're trying to find something that they're not sure exists.

Issam Elbaytam: Exactly. That's perfect description, yeah. You're right.

Richard Campbell: I'm a student of Edward Tufte who was the visualizations guy and Ralph Kimball who is the data analytics guy way at the beginning and of course Kimball was the guy who said if you can't return that information every 10 seconds -- he was very conscious of the fact that the speed of analysis was important because what your executive is trying to do is pursue an intuition that something is here, they don't really know what it is, so that dynamic part is really key, you need to constantly be drilling into data to be able to pursue the intuition successfully, find the exception in the data.

Carl Franklin: You mentioned Tufte. Don't forget about Dr. Codd.

Richard Campbell: Codd, yeah. That's the funny part about relational databases of course they were built to Codd being the father of relational database. He was the guy who really said the whole point of doing this is reporting.

Carl Franklin: Right.

Richard Campbell: Even though that's not what we ever used it for.

Carl Franklin: Yeah.

Richard Campbell: I found a great parallel here for Data Dynamics that your original product was an analysis product and people didn't really understand it. It wasn't until you made a reporting product that everybody got comfortable with you.

Issam Elbaytam: Exactly, yeah.



Richard Campbell: And to this day, I think still your most famous product.

Carl Franklin: Yeah.

Issam Elbaytam: Yeah. Actually, a lot of people don't know the name of the company behind Active Reports. "Yeah, of course, I know who you are now."

Richard Campbell: Yeah. "You're the Active Reports guys."

Carl Franklin: Right.

Issam Elbaytam: Exactly. So, that's actually why we're changing some of the naming convention for our product names. Now, we include the company name with the product. So, we call it Data Dynamics Analysis. That's the name kind of like Microsoft Word. That's the strategy. So, we're trying to bring the name of the company to the forefront and that's I guess sometimes you're going to struggle with when you have a popular product like Active Reports.

Carl Franklin: Do you know how to build web 2.0 AJAX applications with web 1.0 components? Right. You just can't. In order to have next generation web apps, you need next generation components and that's exactly what our friends at Telerik have for you. Their upcoming product codenamed RadControls Prometheus is a huge pack of web controls built on top of Microsoft ASP.NET AJAX, which will add previously impossible performance and interactivity to your next project. Just listen to this. The new controls mirror the ASP.NET AJAX API, so development is straightforward. Client scripts are shared, so loading time is pretty much instant and if you just set a couple of properties, you'll be able to automatically bind to Web Services for even more efficient operation. After all, the facts speak for themselves. The new RadEditor for ASP.NET AJAX loads up to four times faster than before. Similarly, RadGrid handles thousands of records in mere milliseconds, but, again, it's best to try it for yourself. Visit telerik.com/aspnetajax and download a trial. And don't forget to thank them for supporting .NET Rocks!

Now, I notice you have a beta of Analysis out. Can you measure the popularity of this? Is this going to be a popular product do you think?

Issam Elbaytam: We think so, but I think the education aspect is still there.

Carl Franklin: Yeah, required.

Issam Elbaytam: People now understand more about data analysis and OLAP and business

intelligence because Microsoft really pushed that and I guess brought it to the masses, if you will, with Analysis Services. The problem with Analysis Services is there are some awesome server products that really does not have good frontend tools and I believe with Analysis, we've created an incredible, incredible frontend for Analysis Service or, for that matter, any type of relational database because in addition to accessing the analysis, the pre-summarized cubes in Analysis Services, we also do our own aggregation if we have to. So, if you point this at a SQL Server, we're going to pull the data and actually analyze it yourself, so giving you some of the functionality that actually Analysis Services provides to you and you don't have to go through the idea of creating cubes and loading the transactional data into multidimensional data and so on.

Richard Campbell: Now, given that that's so, why would I ever bother with OLAP server then?

Issam Elbaytam: Well, the amount of data, really. I mean if it's a simple application, let's say you're using SQL Server, if you have millions and millions of rows, we're not going to be able to handle that for you. We're not going to be able to pull it out of SQL Server and put it on the client where we handle everything in memory. I mean this is .NET after all. I mean we're going to show millions of rows and try to summarize them. It's going to take time, but if they are pre-summarized in OLAP server and you already have that and invested in it, then of course, you could easily we get it at an instant because we don't have to do any aggregation.

Carl Franklin: It's really worth repeating what you just said before that Richard commented on, which is that you do not need an OLAP cube, you can use relational data, and that SQL Analysis Services, while it's great on the backend, doesn't really have the frontend that you offer here.

Richard Campbell: For a long time, all Microsoft was really offering as a frontend to OLAP server in the early editions was Excell.

Issam Elbaytam: Exactly, yes. The only thing they had is recently is I think they bought ProClarity.

Richard Campbell: Yeah, they bought ProClarity and now they're coming with -- I can't remember the name of it now, but it's like...

Issam Elbaytam: PerformancePoint.

Richard Campbell: Yeah, PerformancePoint, that's what I'm thinking of.



Issam Elbaytam: Yes, but even that, it's not really -- again, you see, they ignored the developer market.

Richard Campbell: Right.

Issam Elbaytam: Which I think really is a very grave thing to do because, really, developers are the closest to the data.

Carl Franklin: Sure.

Issam Elbaytam: I mean they were the ones that built the system that put the data there in the first place.

Carl Franklin: Yeah.

Issam Elbaytam: So, you should not alienate them when you're trying to get the data out.

Carl Franklin: Do you think most of your competition, these big ticket item packages, since they are not geared towards developers, do you think that you actually have an advantage? Well, you do have an advantage because you're not competing with other developer products. You're competing with them, but does it make it really difficult to sell? I mean because most developers aren't going to understand where the competition is and most IT people don't know the developer market. It must be a real challenge.

Issam Elbaytam: Yeah, I mean you hit the nail in the head really. The problem is that we're trying to sell a product that an IT person is typically in control of -- we're pushing it to the developer. So, what we're hoping is that once the developer sees it, he will take our case to the IT person and say, "Okay. So, you're trying to go and buy, let's say, Cognos or you're trying to buy business objects OLAP and data analysis and all that stuff. Well, how about if you go and buy a component for \$14.99 and I build a shell around it and there you go. You'll save thousands and thousands of dollars and I could customize it in any way you want. I could make it look like anything you want because it's part of the application." That's the brilliance of it. As a component, you could take it and it becomes part of the application. So, if maybe your user is advanced user, you just give him the raw feel of it. If your user is a simple end user, then you could give him maybe wizards around it. You could guide him through. I mean you customize it any way you want rather than an application that pretty much looks the same. Whether the user is advanced or not, it looks the same. That's the problem with things like all these tools that you see out there that basically it's an IT person purchasing it and trying to plug it into the data and presenting the same frontend for everyone

in the organization. That does not really necessarily meet everyone's need.

Carl Franklin: Now, is any code required?

Issam Elbaytam: No, code is not required really. I mean basically you've seen it, how we drop the control, connect it to...

Carl Franklin: Yeah.

Issam Elbaytam: Code is not required. However, code is there if you want to. So, let's say you want to change the look and feel or you want to control certain things that a user can or cannot do or should or should not do, you can.

Carl Franklin: But you're basically setting properties to turn features on and off is what you're talking about.

Issam Elbaytam: Exactly. We have a very, very API to customize it if you wish.

Carl Franklin: Yeah. There's never any code to, say, return data, but...

Richard Campbell: So, I'm thinking through a couple of scenarios that I've been through on this, which is often, people don't know they need data analysis. They're looking for reporting tools and, really, because of that iterative behavior, you realize what they're really doing is analysis. So, they're not about to drop \$50,000 or \$100,000 on Analysis because it just hasn't surfaced that much. This looks like an inexpensive way to get started.

Issam Elbaytam: Exactly.

Richard Campbell: And then when they realize, "Wow. This is the best part of the app. I love this analytics bit, but it's not fast enough."

Issam Elbaytam: Sure.

Richard Campbell: Then, because it's querying this massive set of data back to the SQL Server, then you can make the pitch for "we need to buy an OLAP server" or "we need some other additional services to speed things up."

Issam Elbaytam: Exactly, yes. I mean this will get your feet wet. The tool itself is so powerful even in a small dataset. Just the experience is so -- I mean it's amazing. I mean Carl was here and he saw how you could -- you could just go nuts with it.

Carl Franklin: Yeah. My jaw hit the floor.

Richard Campbell: Well, that's the experience I've always had when I put in an analytics tool that was properly configured with the right set of data in front of an executive. They go mad.

Issam Elbaytam: Exactly.

Richard Campbell: That's when you know you've hit it because just off they go, they're gone. You can't even talk to them anymore because they're busy drilling into data and learning something about the business.

Scott Wileke: Just to reiterate that, when we go to shows, it's a very common thing that developers might not even -- they don't take a whole lot of interest at first, but a lot of times they go and they bring a manager back or a manager might be there and they might go grab a developer and say, "Hey, man. Look at this. Wouldn't this be great?" It's really after they see that interactive experience that they're really sold on it, but I think if they just see a line item somewhere, another pivot table or something like this, there's just really no interest there, but when they see this product is working, actually, in front of them or better yet have a chance to use it themselves, it just really knocks people out. I mean they just really go nuts with it.

Carl Franklin: And this isn't the only product that you showed me. You showed me a whole new codebase for reporting. Tell me about that.

Issam Elbaytam: Yes. That's the product that Scott actually spearheaded. What we're trying to do really is -- really, Active Reports as you know is a very hardcore developer as you know and developers just love it, you know. They are "I want to get down into code and then customize and then do this." We also had a segment of the market that we were not reaching and we are aware of that. There are some developers that simply are not interested or they don't want to spend as much time in the trenches dealing with the raw power of Active Reports. So, we thought a reporting engine that has a lot more in the box while keeping up with the history and the legacy of Active Reports would be appropriate and would open some new doors for us, basically. So, we've created a whole new reporting engine that has a lot of features, I mean that are so powerful and so unique actually for a reporting product. Data Dynamic Reports is the new product and it shares a lot of the core with Data Dynamics Analysis. So, basically, we've created a single core and out of it we grew two separate products and we're adding even more to that suite of products. So, Data Dynamic Reports has very unique features such as master pages. If you know about master pages in ASP.NET, that's a very, very popular features in ASP.NET where basically you design a master page and then all your pages build on top of it

similar to inheritance and object-oriented languages, but the idea in the report is also very similar. You create one master report and then all your reports basically sort based on that report and if you go and change the master, all your reports change.

Carl Franklin: Yeah.

Issam Elbaytam: So, imagine in Active Reports, I know some of our users would build hundreds of reports in a single application, but they have to repeat so many things and if the data source changes, for example, they have to go into each report and maybe have some dynamic code behind the ports that modifies the connection or does certain things. With master reports, basically, you just go in and change that single report, switch it maybe from your test database from your production database and all the reports instantly change. You want to change, for example, the header of the company or you want to change the footer to add some new page numbering scheme, all of your reports will change automatically. So, that's a very, very powerful feature. It's a great time-saver. Another one is themes, basically another idea similar to CSS in HTML. So, we define themes that you could use fonts, images and colors that are shared across many reports. You change the theme or you flip themes on different reports to give them a different look or something like that.

Carl Franklin: So, does this new product have the same core features as Active Reports .NET plus these new things or is it geared toward separate...?

Issam Elbaytam: It does, but there is a significant difference though in this style of reporting. As you might be familiar with the idea of banded reports, section-based reports, where basically you open the report, you start with a blank design, and you're looking at bands. Bands are basically like your report had a band or section and your group header and group footer and detail, all these we call them sections or bands or areas depending on the product you're using. So, that is a very linear and vertical design of a report basically. So, you're starting from the report header, then page header, then group header, and then details and you start to close each one of those that you opened up. Data Dynamics Reports is completely different. It uses something called region-based reporting where basically you're looking at a blank page and you drop different regions on it and each region, think of it as almost like a completely separate report and these regions are very powerful. One of them is a table, another is a chart, another is a list, another is a banded list. We also have a calendar. These basically are self-contained data layout regions, okay? So, they bind to a data source and they take the data and lay out in a certain form. That form could be tabular or a table, a chart, graphical, or a calendar for example if you have dates

that you want to throw on a calendar and print it out. So, it takes all these things and you could place them anywhere on the report. This is a different way of laying reports out and it has a lot of power, but is also different from the way Active Reports does it.

Carl Franklin: Oh, that's interesting and you say Analysis uses this new engine?

Issam Elbaytam: Analysis, the core data engine, is the same. So, the aggregation, the summary, the collecting data, connecting to data sources is the same between the Data Dynamics Reports and Data Dynamics Analysis and then the two fork. One takes this data engine, all the aggregation, and presents you with a dynamic view, interactive view, which is Analysis, and the other one presents you with a static view that you could print, export, and do all kinds of stuff, you know, the typical reporting features and that is Data Dynamics Reports, but at the core level, internally, they have the same engine.

Carl Franklin: Cool. So, is the reporting engine available yet or is it still in beta?

Issam Elbaytam: Yeah. It has been actually available for almost a year now.

Carl Franklin: Okay.

Issam Elbaytam: Less than a year, but, yeah, it is available actually and it is being sold on our website. We're having a difficult time basically differentiating between Active Reports because, really, some people say, "Well, I love Active Reports. Why should I switch?" The answer is really you don't have to, you should not if you like Active Reports and you are using it. Really, with Data Dynamics Reports, we're targeting a different type of people that we were not getting with Active Reports, the ones that are not hardcore developers. I'll give you an example. If I wanted to create a highlighting, for example, I want to highlight a certain number in Active Reports to make it stand out. Let's say if the quantity is below a certain level, I want it to appear in red. In Active Reports, you could do that so easily and people love it. You jump into data as a format event and you write some C# or VB code and you say "if the quantity is less than 50, make it red." Well, in Data Dynamics Reports, we don't have that. We don't have event. There is no code behind it. Instead, you write an expression. So, you go to the quantity and you go to the color and you say, "Okay, you write an expression, a logical expression that says if quantity is less than 50, then red; otherwise, black." You know?

Carl Franklin: Right.

Issam Elbaytam: So, it's a different way of thinking. Also, we put a lot into the box. So, in Active

Reports, you have to do more work, but you get more power. With this, you do less work, but you get more features. For example, we have a set of dashboard controls. For example, a bullet chart, a sparkline, a Tufte actually invention. You mentioned that with the...

Carl Franklin: Right.

Issam Elbaytam: Sparkline is a word, they call it a word-sized graph or chart that basically gathers a lot of data and puts it in a very small chart, line chart or bar chart. So, we have sparkline, we have bullet chart, and we also have visualization basically methods to visualize the data like the stuff that you see in Excel 2007, for example, like color scale, data bars, icons that make your data stand out. So, all of these things are built into the report. You don't even have to write a single line of code to get them. It's got all the stuff that AR has from report designer to RDL API, these types of things, you know.

Carl Franklin: That's great. What is the relationship between this and, say, SQL Server Reporting Services?

Issam Elbaytam: Okay. The only thing that they share is basically the RDL file format. As you know, when Microsoft created Reporting Services, they created an RDL file format that is common and they basically made that available so that others could use it. So, we use the same format, but what we built is a whole new engine from scratch ourselves and we use the same format that they use, but we expanded it quite a bit with these controls that they don't have and these features that they don't have as I mentioned the master pages, the master reports, the themes, the dashboard controls, the calendar control, bar code, banded list, all kinds of stuff that we've just added. So, it became double the product that Reporting Services is, but it's based on the same RDL file format that they have. It's XML-based and things like that.

Carl Franklin: And if you use Data Dynamics Reporting with SQL Reporting Services, are you going to get any benefit there if you use the two together?

Issam Elbaytam: Everything that Reporting Services does, we do.

Carl Franklin: Okay.

Issam Elbaytam: So, there is really no added benefit to...

Richard Campbell: So, is the implication there that I could take RDL generated by Reporting Services and then use it to migrate to Data Dynamics?

Scott Wileke: If you have Reporting Services already and you've made that investment in creating a bunch of reports in Reporting Services, then the big difference is that in Data Dynamics Reports, you can import them all and they were drawn as is.

Richard Campbell: Right. So, I don't have to start over. I can move all my existing Reporting Services reports to Data Dynamics, but the reverse is not necessarily true because you've extended RDL. I couldn't take a Data Dynamics report and put it into SQL Reporting Services?

Scott Wileke: Well, we do generate standard RDL, but it's exactly right what you said. I mean Data Dynamics Reports has things like a formatted text control that you can put on there so you can embed HTML in the reports. So, if you have some form letters or something like that you want to be able to put in the report, you can do that in a Data Dynamics report. If you were to put that in Reporting Services, they don't have such a control like this. It's not going to work. The same with the calendar and the dashboard control that Issam mentioned, the master pages, the themes. There are many things like this that are supported in Data Dynamics Reports and, really, very powerful and natural features to use in the products that Reporting Services just doesn't have at all.

Richard Campbell: That's awesome. It's nice to see a clean path because that's always the important part here is, you know, we've made commitment to other tools and then we run into their limits and want to do more when we see an opportunity to do more. Part of the resistance to move that tool is what happens to my existing investment. So, I'm really appreciative that you guys made it easy for me to move from one tool to the next.

Issam Elbaytam: You know; one great thing that we've done with Active Reports and have been very popular is that the Professional Edition, we included our designer itself. We actually made the report designer a control. So, you could go and build a report designer right within your application. So, that has been a very popular feature and a lot of people bought the Professional Edition for that.

Carl Franklin: Right.

Issam Elbaytam: Basically, "I built all the reports for the customer, but he wants more so I'm just going to give him the report designer. Can I give him that?" At first, we used to say no, then we thought, well, why not just make it a control and let them customize it and control it. So, the beauty of Data Dynamics Reports is that we did the same thing. So, we took the report designer and we also included as a control royalty free that you could actually allow your users to

create reports or modify reports right within your application.

Carl Franklin: Right.

Issam Elbaytam: That is something that Reporting Services of course does not have either. Instead, they have some end user report designer. I forget what it is, but it does not share a lot with Reporting Services and this has not been very successful, but this one, I mean, we give you a full-blown report designer that is region-based right within your application if you want.

Carl Franklin: And, you know, the old saw is distribution problems aside, that's what people really liked about Access.

Issam Elbaytam: Exactly.

Richard Campbell: I think we're all coming to this exact same point here, which is one of the focuses you've always had is integrating the reporting and the analysis as part of the application where so many other products, that's a completely standalone thing.

Carl Franklin: Yeah.

Scott Wileke: That's right. That's a really good point. Issam mentioned that end user report designer, that's a key point that you can embed it right into your application. So, if you have an accounting application or something like that, not a foreign separate environment that the user goes into to create reports is actually part of this accounting application. So, he can tailor it with wizards or other integration points into the application that really make it feel for the end user like they're not switching contacts, definitely not moving to another application, let alone it looks the same and go back to where they were in their application very easily.

Carl Franklin: Speaking of Access, Richard, that Access Upsize Wizard that you guys have?

Issam Elbaytam: Uh-hm. Yup.

Carl Franklin: That's coming very handy on a couple of projects.

Issam Elbaytam: Yeah. By the way, we have the same thing also in Data Dynamics Reports. So, not only you could read RDL reports, but we also have the same Access upsizer and the same and Active Reports converters also. So, whatever reporting tool you're using, you could also bring it into Data Dynamics Report. One other thing I wanted to mention about the whole idea of putting the tool within the application. As you know, for a long time, Microsoft has been pushing people. Basically, a good

development methodology is not to really directly access the database within your application. It's to have these different layers and properly breaking down the application into data layer, business objects and all these types of things. Well, when you're using an external tool, so for example you want to bring in Analysis, you want to bring in Cognos, or you're bringing some other PerformancePoint or something in that all these things would require you to break that development methodology that you've been working at and the application architecture because they have to go directly to data, but when these controls such as Data Dynamics Reports or Data Dynamics Analysis are integrating within your application, they can use the same data access method that you are using. So, if you happen to be using business objects, for example, you could run Active Reports against the business object itself. You don't have to say, "Well, Active Reports requires me to go to the database." No. Also, if you're using, for example, arrays or collections of certain things or you want to print a report from multiple data sources, maybe you grab something from XML and then you grab something else from the data and then something you calculated inside your application on the fly, you could get them all together and represent them in the report because the report is living within your application. It understands these things. Unlike if it was an external tool where it doesn't have a clue what you're trying to do.

Richard Campbell: Now, I'm just thinking about the way people operate. I'm a fan of the sovereign custom app, right? The app that owns a chunk of your real estate on your screen all the time when you're at work and that's where this stuff would live, but I'm also thinking about the executive who probably doesn't spend a lot of time on certain sides of the app maybe does want a standalone Analytics app. Now, I disagree with myself already. You know, the biggest thing I like about the integration option is the first thing an executive does when they find a piece of information that's insightful is question its validity, right? They go, "Why is that sales number so high?"

Carl Franklin: Yeah.

Richard Campbell: And most analytic tools, they hit a wall right there. They usually can't get at the data per se and they certainly don't know all the business rules and all those sorts of things, so the usual reaction to that guy finding a piece of data he questions is to phone you as the developer and say, "This data is wrong. Go find out," and then you spend the time trying to prove it.

Carl Franklin: Well, what's great about this is that instead of just getting a number, you see patterns and correlations like, Richard, they have a great analysis tool, the Correlation Analysis tool, that draws

little circles at data points and then when those circles are bigger that means something, when they're smaller it means something, and also when they're coagulated all around, I mean you can really get a sense for the big picture with some of these very simple graphical tools. Great stuff, guys.

Issam Elbaytam: Thanks.

Scott Wileke: And I think something you are saying like if the executive does want an Internet portal or something like this or his executive reporting thing, all of our products work in a scenario like this.

Richard Campbell: I'm sure you have all the choice in the world you're customizing. I'm thinking really...

Scott Wileke: Exactly. That's a really key point. I mean since they all work in ASP.NET, it is a common scenario that unfortunately programmers don't get to run the company all the time, so when an executive tells him that he wants it on this site, this Internet site that he goes to every morning, at least you have that option with all of our products and it's very common for a customer to do just that.

Richard Campbell: Yeah and I like the fact that you could use the same set of tools as a standalone web app could also be embedded in the sovereign app on the desktop. I think one of the angles I was trying to drill into here is that I can see where it would be incredibly useful to have pieces of the app, of the sovereign app available to the executive so that they can drill in on the data themselves. They can find out what was that huge sale that made that stand out figure in the analysis side of things themselves and then say, "Is that sale real? I can now go to the salesperson rather than going to an IT guy or to a developer to try and drill down to that information."

Issam Elbaytam: Exactly, yup.

Carl Franklin: All right. So, what's on the burner for you guys? I mean this is some great stuff that's coming out. You're looking forward obviously to the next thing. What's next?

Issam Elbaytam: Exactly. So, what we have now is between Data Dynamics Reports and Data Dynamics Analysis, I told you that they share a core engine and we're going to build on that quite a few things actually. We're going to try to basically fill up the whole spectrum of anyone that needs to do data analysis, business intelligence, reporting, information delivery, all these types of things, we're going to create products that reach into each and every segment. With Active Reports, we have covered the hardcore developer. With Data Dynamics Analysis and Data Dynamics Reports, we're trying to bring that closer to people who are not interested in that much



coding and now we're going to even take it further by creating a very powerful end user report designer that basically lives within the browser or within the application that is targeted, tailored even more towards end users. They don't have to think about data. They think more about models and entities and things like that. Then the next step is really to start scaling this to more enterprise level, you know, with reporting distribution and servers and these types of things. They're all going to be centered around the core data summarization and aggregation engine that we've built in Active Reports and Data Dynamics Reports Analysis.

Carl Franklin: Oh, that sounds great. I can't wait to see that. Well, we're coming to the end of the show here. I can't take you enough for being our friends and sponsoring our show and having great products that we can believe in and talk about. I'm sure the listeners are really as interested in it as we are too, so thank you guys.

Issam Elbaytam: Thank you very much. It's been an honor. Thank you.

Scott Wileke: Thank you.

Carl Franklin: Issam, Scott, thanks again and we will see you next time on .NET Rocks!

[Music]

Carl Franklin: .NET Rocks! is recorded and produced by PWOP Productions, providing professional audio, audio mastering, video, post production, and podcasting services, online at www.pwop.com. .NET Rocks! is a production of Franklins.Net, training developers to work smarter and offering custom onsite classes in Microsoft development technology with expert developers, online at www.franklins.net. For more .NET Rocks! episodes and to subscribe to the podcast feeds, go to our website at www.dotnetrocks.com.