



<http://www.dotnetrocks.com>



Carl Franklin

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!



Richard Campbell

*Text Transcript of Show # 185*  
(Transcription services provided by [PWOP Productions](#))



## Mark Miller on Discoverability July 13, 2006

### Our Sponsors



**Developer**  
EXPRESS

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



innerworkings

<http://www.innerworkings.com>



telerik  
radcontrols  
for ASP.NET

The preferred UI suite for ASP.NET



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



(Music)

**Geoff Maciolek:** 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)

**Geoff Maciolek:** Hey Rock heads! Stop expecting me to come up with jokes at 6 AM and listen up; its time for another stellar episode of .NET Rocks! – ‘The Internet Audio Talk Show for .NET Developers’ with Carl Franklin and Richard Campbell. This is Geoff Maciolek, here to announce Show #185 with guest Mark Miller, recorded live, Thursday July 13<sup>th</sup> 2006. .NET Rocks! is brought to you by [Franklins.Net](http://www.franklins.net)– ‘Training Developers to Work Smarter,’ and now offering a whole suite of onsite and remote classes in .NET 2.0 Technologies, online at [www.franklins.net](http://www.franklins.net) and by telerik.r.a.d. controls, the more comprehensive suite of components for Windows Forms and ASP.NET web applications,” online at [www.telerik.com](http://www.telerik.com); support is also provided by Developer Express -- crafting first class tools, frameworks and controls for the .NET developer -- improve your experience online at [www.devexpress.com](http://www.devexpress.com) and by CoDe Magazine, the leading independent magazine for .NET developers, online at [www.code-magazine.com](http://www.code-magazine.com) -- and now, the man crazy enough to trust me to make this intro funny, ha ha ha..., Carl Franklin.

**Carl Franklin:** Thank you and welcome to .Net Rocks! Unfortunately Richard Campbell is not with us today, he is...

**Mark Miller:** You know what, it’s about time Carl you came to your senses and made me co-host of this show. Hey, who are we interviewing today, I heard it was some guy named Rory Calhoun or something like that.

**Carl Franklin:** It’s Rory Blyth.

**Mark Miller:** Ah...Rory Blyth of course, I remember the young lad. I believe I interviewed him in one of my previous hosting gigs on this great .Net Rocks show of ours Carl.

**Rory Blyth:** Hi Mark! Excuse me Mark.

**Mark Miller:** Hi Rory, so Rory, I am sure our listeners want to know all the latest gossip details, like how tall are you, do you have any allergies? And where exactly do you stand on romantic walks in the park.

**Rory Blyth:** Mark!

**Mark Miller:** Do you really enjoy them like it says you are on your online profile on [hotyetstillsingle.com](http://hotyetstillsingle.com) or do you simply consider them as a means to justify the end?

**Rory Blyth:** I am actually pretty sure I am not listed on [hotyetstillsingle.com](http://hotyetstillsingle.com).

**Mark Miller:** Oh really! So you expect our audience of astute podcast listeners to believe that you are not indeed Mr. N. Bonaparte self described as tall, dark, handsome and extremely generous to the right lady and “it’s not only my nose that’s big -- and looking for classy ladies who would like to wear shiny clothes, chew gum and hangout near street lamps after dark?”.

**Rory Blyth:** Oh, oh! Well yeah, I mean, well no, that is me.

**Mark Miller:** Hey Carl, how do you like my hard-hitting interview approach so far, eh! Pretty awesome, eh! Its just like 60 minutes and I am that old guy here on this show and you are like the production assistant -- why, you are getting me some coffee, we just surprised Rory in his office and guess what, he is just shredding documents that prove he was involved with the great doughnut shortage of 1987. I told you, you would never regret the day you finally decided to make the Millinator co-host of this show.

**Carl Franklin:** Actually Mark, Rory is the co-host this week and you are the guest.

**Mark Miller:** Sorry?

**Carl Franklin:** As you have always been the guest.

**Mark Miller:** I was never the co-host?

**Carl Franklin:** No.

**Mark Miller:** What about the host, I was the host once right?

**Carl Franklin:** No.

**Mark Miller:** Oh!

**Carl Franklin:** So Rory, my apologies for that, I’ll make sure we edit that out of the show and as I said, Richard Campbell is out of the country, he is climbing Mount Fuji in Japan and will be returning back to his homeland shortly. And before I welcome Rory as my co-host I’d also like to say that we were scheduled to have a product announcement from Microsoft this week and some last minute mix-up in Redmond with blue



badges prompted us to insert this show into the schedule. So, next week, God willing, we will have the announcement -- the product announcement for Microsoft. So I would like to welcome Rory Blyth back as my temporary co-host; he is here recording some stuff in the PWOP studios with me. How are you doing Rory?

**Rory Blyth:** I am doing fine, um, considering that you ever working me and sticking me inside of a little tiny sound booth on a really hot day where its really sweaty, humid and hot.

**Carl Franklin:** You can always ask for your money back, pal.

**Rory Blyth:** Yeah, give me money.

**Carl Franklin:** Well, it's good to have you back; I know that a lot of listeners are surprised to hear your voice.

**Rory Blyth:** I know I have looked at the charts, I have seen the line that goes straight down as soon as I left I know how it works, so.

**Carl Franklin:** (Laughing) And, of course we all know that Mark Miller is the guest, "The Guest, Mark."

**Mark Miller:** Sorry about that Carl, that little faux pas at the beginning there.

**Carl Franklin:** What's on your mind?

**Mark Miller:** Oh... you know, nothing really -- I got nothing Carl.

**Carl Franklin:** All right so have you downloaded anything cool lately?

**Rory Blyth:** (Laughing).

**Mark Miller:** Nothing, I have downloaded nothing. I unplugged myself from the Internet, I have just been sitting here looking at my screen; actually...

**Rory Blyth:** This guy is a jerk; I am serious.

**Mark Miller:** Yeah, all right.

**Carl Franklin:** Before you turn off the show listener, this isn't a normal show for us but we do have some content. Mark, you have been telling me that you are very concerned with discoverability issues.

**Mark Miller:** Yeah, we can talk about discoverability. Discoverability is something actually that I have recently discovered as a very

big deal. You know, it kind of gets to this point of how hard you work on the code and how passionate are you about it -- and if you are, and you write a feature that nobody uses -- your customers don't use, then you essentially spin your wheels wasting your time right?

**Carl Franklin:** Yeah.

**Mark Miller:** Your customers don't know it and you know, they are not as deeply in bed with your product because they are not using the features that you work so hard on. So, you know, one of the things that I realized is that discoverability is very-very key -- it's very important to customer satisfaction.

**Rory Blyth:** Hey Mark, hey Mark, just a minute here right, this is Rory -- and you know I am stupid right, and I haven't been on this show a long time. So could you just like sum up discoverability in one short Mark Miller, kind of, crack-inspired paragraph?

**Mark Miller:** Sure man. I need some discoverability man. Yeah, discoverability is essentially the clue that tells your customer that the feature is there -- that's discoverability.

**Carl Franklin:** How easy is it for them to figure out where the feature is; nobody reads manuals anymore.

**Rory Blyth:** Right.

**Carl Franklin:** Nobody is going to read the manual, nobody is going to read the help file, they want to just look at the software and figure out how to operate it.

**Mark Miller:** Right. No you're right, nobody uses -- nobody goes into the help; it's a last resort because it takes so long and it really, it disrupts the flow, you know, realize there is already a flow disruption at the time there is question about the feature or something that they want, also times they may not even know that the feature exists, and they may not even be in the mood to explore. Menus, I think were probably one of the most significant discoverability inventions that has occurred in our field... right?

**Rory Blyth:** In fact, before, people had to just kind of guess what the application did, right?

**Mark Miller:** Well, they had to read the manual right, you know what are the commands, or they had do the /? at the DOS prompt to see what the help was -- to see what it parameters it accepted right? So menus come in and now somebody can grab a mouse and click on it and get a sense of what the application can do. Tool tips are a



discoverability invention that also is pretty significant, it tells you what the button underneath the mouse does, right?

**Carl Franklin:** Yeah.

**Mark Miller:** So, tools tips and the mouse those are useful. And you know, we had a product that had a lot of features, but the features were not -- in order to find the features, you had to go to the manual. And with the recent update that came out about two weeks ago, we added a training window that essentially watched as you worked, and based on the context -- the context being, you know, where you are, what you are trying to do or what you have done so far, based on that, it makes suggestions for what is available and it also does...

**Carl Franklin:** I should also say Mark right now, you know, that I didn't introduce you because you have been affiliated with this show for a long time, but in case you don't know who Mark Miller is, he is the -- forgive me if I get this wrong, you are the Chief Architect for the Tools Division of Developer Express?

**Mark Miller:** Yeah, I think that's about right, that's...

**Carl Franklin:** So you are responsible for CodeRush and Refactor architecture.

**Mark Miller:** Right, so I did developer tools and so the product I am talking about is CodeRush, which is -- and one of the things that we added were these things -- were dynamic templates. And dynamic templates are kind of, it's kind of like a mini language I guess you would call it, strictly for writing code. And I guess it sounds a little scary, but we created this training window to essentially show the language as it was available to pieces as you were right there and the response from customers was really remarkable. In general, enthusiasm was very-very high and we had a lot of people talking about great features that were in the product before that they were just now discovering for the first time.

**Rory Blyth:** That's actually I just want to jump in and say that -- you know I had taken a look at CodeRush before, like the first few times I had met you and when Scott Hanselman once wrestled me to the ground and forced me to watch a CodeRush demo at TechEd.

**Carl Franklin:** Me too actually.

**Rory Blyth:** And it looked really interesting but when I actually installed it on my own machine and started trying to use it, I realized that I was going to have to read the manual; and honestly, I

never started using CodeRush specifically for that reason. So you are saying that if I get this new version of CodeRush and I give it a shot, then its actually going to help me out of it and its going to give me...

**Carl Franklin:** I am not clear, quite, on the training window thing what is that...?

**Rory Blyth:** Yeah, I want another more because I am going to give CodeRush another shot here.

**Carl Franklin:** What do you -- is it like a video window or is it something that just.

**Mark Miller:** It's an HTML Window.

**Carl Franklin:** Okay.

**Mark Miller:** And, for example, if you are inside an empty file, it says -- it's got an entry called "Types" up at the top and next to -- underneath it its got some -- its got the letter "C" and to the right of that, it has is the word "Classes" and underneath that it has got the letter "S" and to the right of that the word 'Struck'...

**Carl Franklin:** So it's almost like dynamic help then for CodeRush.

**Mark Miller:** Yeah, I never found dynamic help in -- I shut that off so fast, but it was just...

**Carl Franklin:** What I mean is dynamically changing uh, tips.

**Rory Blyth:** Its contextual.

**Mark Miller:** It is contextual and it changes; so you type in "C", it will say, 'Press the spacebar to expand and to create a new class.'

**Rory Blyth:** Okay, now that's pretty cool actually.

**Carl Franklin:** Yeah that's factual.

**Rory Blyth:** And that's exactly what I need because I was watching Scott Hanselman and he can just -- I mean, to him CodeRush was a video game.

**Mark Miller:** Right.

**Rory Blyth:** He just hit buttons and arrows would shoot up and cool things would happen, and I wanted to be cool like Scott, and then I sat down and I felt sort of like, you know, a one year old trying to learn how to play the piano, you know, so.



**Mark Miller:** Yeah well, I am -- you know, I can do the same things Scott can do and I still feel I want to be cool like Scott.

(Laughter)

**Rory Blyth:** Yeah, so that actually sounds really cool.

**Carl Franklin:** Yeah, it does.

**Mark Miller:** Scott Hanselman has got -- he has got a cool gene that I don't have.

**Rory Blyth:** Well, he has a special part of his brain that just makes him a totally different kind of human being anyway, so that's Scott.

**Carl Franklin:** So, can you limit or do you have control over how much stuff gets shown in that Window because obviously you could throw everything at somebody and that would be just as overwhelming as it...

**Mark Miller:** Right... Well that was the other trick in this, and that's noise reduction is how do we get it, so just the essence is there and then there is an...

**Carl Franklin:** I like that term.

**Mark Miller:** What, noise reduction?

**Carl Franklin:** Yeah.

**Mark Miller:** Now, noise reduction, that's exactly what this is all about Carl, I mean, how many times have you brought up IntelliSense looking for an event and you have had to scroll past all of the Properties and Methods but all you want is the Event, right, there is not noise reduction there for you.

**Rory Blyth:** Right.

**Carl Franklin:** Right. And it could be just as bad to navigate through an endless hierarchy of crap too, so.

**Mark Miller:** Yeah and that's -- yeah, so noise reduction is a good thing, noise sucks, it slows people down, and that's why people avoid the help because when they go into the help, its huge; and they got to go search through it.

**Carl Franklin:** Yeah.

**Mark Miller:** So that's why they use that as the last resort. So, a lot of thought went into, you know, what's the essence, what's the most important thing to teach, you know, the person who would have this window up, right, the person

who is not going to go into the help. What's the most important thing to show them at this particular point in the code?

**Carl Franklin:** That's not easy.

**Mark Miller:** One of the things we also did is, we went into the product and we added more context checking things -- like, as an example, one of the features we have is "Selection Inversion" which will invert a bunch of assignment statements, swap them left and right.

**Carl Franklin:** Yeah, I have needed that.

**Mark Miller:** So, we added a context into the product that said, hey, do we have a selection that is -- that qualifies for a "Selection Inversion" - - and if we do, then we show the key binding for that -- and it's in the list.

**Rory Blyth:** Does this also work with Refactor?

**Mark Miller:** No, not yet.

**Rory Blyth:** It could, easily right.

**Mark Miller:** I say not yet like its going to -- it might. And Scott Hanselman actually made that request because he wanted to see the refactorings that were available; we actually created a plug-in for the Logitech G15 Keyboard which has got a little LCD display on it -- that actually shows you the Refactors that are available so I...

**Rory Blyth:** On the keyboard!

**Mark Miller:** On the keyboard, right...

**Rory Blyth:** Wow!

**Mark Miller:** So you don't have to bring up the menu, you can just look down which is...

**Rory Blyth:** Wow! That is really cool and its great.

**Mark Miller:** It's not bad. Now this whole idea of being...

**Rory Blyth:** It's not bad.

**Mark Miller:** Well, it's not bad. The best thing is a full color display or just more monitor space, but think about what's going on there; the person is curious to know what refactorings or what features are available in your product right? So their action, the energy they put into it is essentially to move their eyes a few degrees to look down. And that's great accessibility. Right? That's the easiest possible thing; they don't have



to reach for a mouse, they don't have to hit a key, they just look down -- and that to me is really where a lot of our future direction is going to be focused in, is how do we figure out what question is in on your mind and answer that question for you in this particular area so that if you have a question as you are writing code, you just look and you see the answer.

**Carl Franklin:** This portion of .NET Rocks! is brought to you by our good friends at InnerWorkings; inside Visual Studio, you can take any number of programming challenges on topics as varied as .NET Security, Message Queuing, XML and Web Services, Data Binding and Object Oriented Fundamentals. Once you have completed a drill, the developer interface will judge your code, giving you a thumbs-up or thumbs down, and if you fail, it shows you where you went wrong and where you can improve your skills. So, I encourage you to go to [www.innerworkings.com](http://www.innerworkings.com), download the free stuff, check it out, take it for a test spin and tell me this isn't the coolest thing you have seen in a long time.

(Music)

**Carl Franklin:** You know, as it usually happens, when I talk to you and you have some new great thing to share, my mind starts thinking about -- let's take this to another place, and what occurred to me that might be of value is if there could be a sort of, standard side-car monitor that might be a small -- maybe five or six inches square or something like this.

**Rory Blyth:** Meant specifically for....

**Carl Franklin:** Meant specifically for discoverability that you could program to with an API and Windows.

**Mark Miller:** Yes. Well I think its good, I mean the -- when I start working with the Logitech keyboard, with its display, my -- within a couple hours of programming it, my immediate reaction was, this thing needs to be bigger, it needs to be higher resolution and it needs to be in color. And...

**Carl Franklin:** Yeah. Video help for example, could play there, yeah.

**Mark Miller:** Yeah to some degree; I do agree though that a separate display is built in somewhere or maybe underneath your current monitor -- it does have value, but I think ultimately what we are going to have is, we are just going to have bigger, bigger monitors, and people are just going to set aside a chunk of it and say, this is where I am going to put in my

panel that's going to teach me or show me -- give the contextual information that I need for running this product.

**Carl Franklin:** That makes a lot more sense -- and because obviously bigger monitors you can do what you want with it. But I like the idea of having a sort of a standard API.

**Rory Blyth:** But Carl, you have given away too many million ideas on this show. (Laughter) I mean you really have.

**Carl Franklin:** I should just shut up.

**Rory Blyth:** You should just stop. Because...

**Mark Miller:** That's was a million dollar idea?

**Rory Blyth:** Okay see, Carl?

**Mark Miller:** I just got to take a note man, hold on a second, let me write something down.

**Rory Blyth:** So, can I change gears a little bit here, because I have got a different question about your product, all right? Is that okay Mark, is that okay with you?

**Mark Miller:** Yeah sure. That's fine.

**Rory Blyth:** Can we do that? Okay. Now as somebody works for Microsoft now, and as a nerd and a geek here -- I was about to say a neek -- as in nerd and a geek...

**Carl Franklin:** You just invented a new word.

**Rory Blyth:** I did invent a new word, I'm an elegend. What I wonder about is, I don't know if any other company -- and correct me if I am wrong here, that does so much work on integrating with the tools with the IDE as you guys do. And I am wondering how closely you work with the developer tools division in Microsoft. I mean, how do you guys -- how did you figure the stuff out, because CodeRush is not just like a neat tool, but the way that you guys have it set up, I mean, its even like visually pleasing -- like, its kind of satisfying to watch it being used. I mean, how in the hell did you do that is my question, because I have looked at the model myself and I have worked on writing plug-ins and it's not easy -- these guys that are exposed.

**Mark Miller:** It doesn't actually support a lot of the stuff that we are doing. So, we essentially had to -- the way I like to describe it is create a safety net to make this whole thing work. So it's essentially, I guess, at its crudest descriptive form, you'd say it's a hack. But it's a hack with a



safety net that makes it a stable, releasable, kind of thing. I mean, I say hack, I mean we are using Win32 API stuff. And I mean we are using all approved -- we're not using anything undocumented or any of the crazy stuff like that. But that's essentially what's going on. So, we are doing -- we are just doing a lot -- there is a lot of things that are there and it's a lot of classes or a lot of objects that are working together to ensure that everything is reliable and is working correctly.

**Rory Blyth:** So, you have kind of created your own little managed environment for...

**Mark Miller:** Well we have. Now, both CodeRush and Refactor sit on a third product called DXCore, which is a free product that Developer Express gives away. And it's specifically built for creating plug-ins very easily. So, if you wanted to create something that extended Visual Studio and you wanted to do some simple kind of thing you worked -- well, let me just finish that sentence -- if you want to do some kind of simpler or even sophisticated kind of add on tool, the DXCore would probably definitely be the way to go, I would definitely recommend you go there, because you can do things there that you cannot do through the standard Microsoft ways of extending the product.

**Rory Blyth:** I can imagine yeah, because the COM APIs are rather complex and you have pointed out there is some things you can't even do. So, this product you are talking about, does it expose the managed API or is it all native code?

**Mark Miller:** It's all managed.

**Rory Blyth:** Really? How come I never knew about this?

**Carl Franklin:** You should watch the dnrTV that I did with Mark where he actually builds a -- by request, a plug-in, in an hour.

**Rory Blyth:** That is awesome. That is actually really, really cool. So, now to kind of, get back to my question about how closely you work with Microsoft, do you get to work with the developer tools division and make suggestions to changes in the model. I mean why -- I mean I have always wondered this about our own products, why don't we have like a managed API that we have exposed for plugging into...

**Mark Miller:** You do. You guys do; and that was created -- well its been in the works I want to say for about two years, but the reason you probably didn't hear about it is because it was not -- when people wanted to do the real deep extensibility of

Visual Studio, they use something called VSIP, which is the Visual Studio Integration Program...

**Rory Blyth:** That's what I am talking about right.

**Mark Miller:** ...which is all COM, C++ files all over the place, a real pain to work with, but there was a group inside of that team that was building managed Interop assemblies. However, we started CodeRush before those assemblies became available, so we were essentially on a parallel path, creating managed assemblies as well for the DXCore. So, they both exist; the thing is though is that the DXCore, what's really cool about it, is it uses a component-centric architecture. So, to contrast, if you want to register a command using VSIP, it's a big pain in the butt. There is a lot of code you got to write. But if you want to do the same kind of thing with the DXCore...

**Rory Blyth:** It is just like command component?

**Mark Miller:** ...you drop a component on the design surface and you set properties and handle events and you are done.

**Carl Franklin:** Nice -- useful thing.

**Mark Miller:** So the whole architecture is like that. And so, we didn't really get into that very much on the dnrTV episode that Carl is talking about, because that was really -- that was mostly a lot of code writing is what that was, because it was a fairly sophisticated request.

**Carl Franklin:** Mark, let's get back to discoverability for a minute; and specifically, I want to know -- you obviously are the architect, you're the brains behind this thing. Did you look at other software and have you looked at other stuff and did you have any critique of, "Oh! That's horrible" or "Oh! That's good," did you get some ideas from other -- from what anybody else is doing. For example, what do you think of Office 12 with a new ribbon -- talk about discoverability, there is a whole show right there, right?

**Mark Miller:** You know, uh, the answer is no, I have not really done a lot of looking at other places for this. In fact, if anything, I found that you are seeing some of the stuff that we have done come out in other applications. As an example, one of the things that we do in refactor is, we have something called 'Preview Hinting' -- and the idea behind it is, when you are either hovering the mouse or you are using the keyboard on a menu item, and you are over a refactoring, like say for example, Extract Method, what you see in the background on the code is essentially mark up, that looks like somebody either grabbed a pen and marked it up...



**Carl Franklin:** It looks like John Madden with a pen you know.

**Mark Miller:** Yes, just show you what the changes are.

**Carl Franklin:** The quarterback comes around \_\_\_ the halfback.

**Mark Miller:** Just show you what the changes are, right?

**Carl Franklin:** Right.

**Mark Miller:** And this is an idea of discoverability where before you commit -- see, normally what happens with a refactoring tool is, you commit, then maybe a dollar comes up and you fill out some stuff, you hit okay and then you see one more dialogue that says, "Here's all the places I am going to change, do you confirm or not?" And you get maybe a preview of the change in that dialogue, and you click okay or not. And we -- I am passionately anti-modal dialogue -- I'll only throw up a modal dialogue if it's something that is a real important thing that is infrequently accessed.

**Carl Franklin:** God bless you, sir.

**Mark Miller:** Okay? So, no modal dialogues, so - - but we want to have discoverability, so instead of forcing the user to choose and then showing them what its going to look like, because I mean here -- let's face it, refactoring serves a lot of them, we've got like, almost 60 in our product right now. And a lot of these have crazy names that are -- if you go with the Martin Fallard approach, so like one of the names like, "replace nested conditional with guard clause." If you don't know what that is, you are not going to choose it, you are going to run away.

**Rory Blyth:** Yeah, I don't know what that is so.

**Mark Miller:** But if you hover the mouse over it and you see a preview hint that shows you, here's what's going to happen, then you are more likely to try it. And so, that's the discoverability thing. Now, I said that we are seeing some indication that these kinds of things are showing up elsewhere in the next version of Office and Word, when you go to change the font, as you move over the font in the menu, the document will change; It shows you what that will look like. And so, then you are seeing that same kind of discoverability, that preview hinting -- it is essentially a tool tip on steroids, is what it is. Now, hopefully I am not going to get sued by anybody who's got a patent on tool tips. But that's what it is; it's functionally the same thing.

**Rory Blyth:** Now, I have seen that ribbon in Office 12 and that's one of the most impressive new features to me. So, you are saying that that's kind of where you guys are going with CodeRush -- or with some of the refactoring I should say?

**Mark Miller:** Well now, I guess where I'm seeing that's where Office is going.

**Carl Franklin:** Yeah, you said Office took nod from them.

**Rory Blyth:** Well I mean...

**Mark Miller:** Didn't I just say that?

**Carl Franklin:** I'm sorry. That's what you said.

**Mark Miller:** Wasn't that excruciatingly clear? To be honest, I have not -- actually, I have never seen the ribbon stuff. Every time it's been shown at a keynote at a conference, I have been like, at the booth.

**Rory Blyth:** See that is actually really cool. It's incredibility cool actually.

**Carl Franklin:** Rory, educate us.

**Rory Blyth:** Okay. It's -- the idea is that, for example now, like, if you want to change the font in Word for a particular paragraph, you actually have to go through and change that font. You got to go up to the combo box, and you got to select the font and you can see in the combo box what the font looks like, but you don't know what its going to look like when its actually applied to your document. Because your document has specific font sizes and other styles associated with it. So, what you do instead, is you get this cool little ribbon -- this cool little window, we get to hover over which font you would like, and it will, in real time, it will dynamically change the font of your word applications so you see what it's going to look like.

**Carl Franklin:** I think Mark just said that.

**Rory Blyth:** Well, yeah but you told me to educate you

**Carl Franklin:** All right educate.

**Rory Blyth:** We'll cut that out of the show, that's not ok.

**Mark Miller:** No, no hold on, I think its good; the thing I was going to say Rory though, is this is great, great stuff; it really is, it's great for the user because it lets them go in and more deeply bond with the application that increases your level of



satisfaction; that means they are going to come around, be your customer next time and they're less likely to leave.

**Rory Blyth:** Sure, my sister is a Mac person and she said that she saw a little bit of Office 12, and she is talking about switching; so, I mean there you go.

**Mark Miller:** There is a context sense of nature to the ribbon as I'm told, right? As you move around, the buttons that are available change, right?

**Rory Blyth:** Which is very important.

**Mark Miller:** So, again that's noise reduction right? Carl, we don't have like, 40 buttons across the top anymore.

**Carl Franklin:** Now the one thing that you notice is, the menus are like gone, right?

**Rory Blyth:** They are not gone -- you can -- we'll kind of like...

**Carl Franklin:** You just sort of buried.

**Rory Blyth:** Have you used IE 7?

**Carl Franklin:** Yes.

**Rory Blyth:** For example, it looks like the menus are gone, but you hit Alt and the menus show up. I am not sure if Office stuff is going to behave in the same way.

**Carl Franklin:** So, Alt is the magic key that nobody is going to know that...

**Rory Blyth:** Right, the thing is, in IE 7.0, a lot of the time you don't need to use the traditional menus; that's something that you would do if you were like, hardcore and you wanted to do something like, change.

**Carl Franklin:** I have had this problem, where I have -- when I first started using IE 7.0 I was like, "Where is the fricking menu?" Like, it isn't very obvious, it's as to where these things are.

**Rory Blyth:** It might be user error.

**Carl Franklin:** No, no, I call it a lack of discoverability is, what I call it.

**Rory Blyth:** Yeah, I am a stockholder and I to refer the company and I call it user error.

**Carl Franklin:** What do think about that Mark, I mean, you got to know that you have to press the Alt button to get the things that....

**Rory Blyth:** I did it by accident. So...

**Mark Miller:** I think Rory is right, it's user error, that pretty much what's striking me was... User error man! Any other questions?

**Carl Franklin:** You obviously don't have a lot of Windows '98 customers, you know, or Windows 2000 users.

**Rory Blyth:** There actually is a reduced simplified menu off to this side for IE 7.0; and if you want the traditional menu you can get to it through Alt. But you can still do everything pretty much with that simplified menu, what, hey, what?

**Mark Miller:** Guys, I just -- I thought it's something important that I want to talk about with discoverability.

**Carl Franklin:** Okay.

**Rory Blyth:** Oh! Cool. Okay.

**Mark Miller:** Well, it has to do with discoverability and noise reduction. So, we created this training window right? The thing is, this training window is essentially -- it's built for the lowest common denominator; in other words, the group, the union of all the customers right? -- Its built for everybody. But what's missing from it is the ability for it to learn from your own usage of the product, "Hey, you know what? He was in a situation, the context was available, he hit the hot key and he hit it within like, .25 seconds; he knows this hot key." Let's now move on to the advanced level stuff. So, it doesn't have it yet, and I really want to build that in...

**Rory Blyth:** That is cool, yeah.

**Mark Miller:** In other words, to make it so that I want to have advanced customers coming back to me, after a couple months of usage and say, "Woah! I just found out this" -- that sort of thing. So, I really want that discoverability panel to continue to reveal little gems as a developer gets better and better using the product.

**Carl Franklin:** This portion of .NET Rocks! is brought to you by Developer Express; crafting first class tools, frameworks and controls for the .NET developer. Improve your experience online at [www.devexpress.com](http://www.devexpress.com).

(Music)

**Carl Franklin:** This reminds me actually of a suggestion that I made that one of my blog readers turned into a little product. It was a couple of years ago, and I had sort of come to



this idea where making user interfaces that have three levels, like Newbie, Intermediate and Expert, and I know that you have that in your UI Mark. But the whole idea was that you could drop this extended provider on a Form and then it gave a Property to all of the controls as to what level it was at.

**Mark Miller:** Yeah cool idea.

**Carl Franklin:** And then simply by setting the level, those things were enabled and disabled -- or visible and invisible. So, just by one line of code, you could sort of crank up the UI. And I remember, I had a whole bunch of people comment on it on the blog, including Allen Cooper who said I was a fricking idiot.

(Laughter)

**Mark Miller:** Cooper said that?

**Carl Franklin:** Yeah. He said you don't get it.

**Mark Miller:** I thought it was brilliant but there were a lot of nay sayers for some reason.

**Rory Blyth:** Cooper is a hack.

(Laughter)

**Mark Miller:** That's right, I am going down, going down on the record for it -- Cooper you are hacked.

**Carl Franklin:** I think you are going to be visited by the guys in the black suits.

**Rory Blyth:** You and me man, Mono and mono. Now, Cooper actually is very good. He has come out with some great ideas that I have tried out for real, and most of them are pretty good. One of them was not as good and I was -- it was one of those things where I was like, just kind of blindly doing it, I was like, you know what, this kind of feels good, feels good, feels good and I did it, you know, and it kind of looked good but I was really more -- I was seduced by the sexiness of it rather than the usefulness of it and ended up cutting it from the product. I just was like, you know what, I was pushing it too hard. So, I am kidding about that, don't hunt me down.

**Carl Franklin:** Yeah. I know. Well, it was interesting that one of the blog readers actually did turn it into a very easy to use DLL Assembly, and it is still up there on my old blog, maybe I will find it and post a link to it.

**Mark Miller:** I think that the future, to me at least, for usability is with applications that learn how well you were consuming them and adjust their

systems appropriately based on your usage. I think there is definitely potential, there is potential there to create a more satisfying experience for the customer.

**Rory Blyth:** I agree; it's got to be tricky to program that kind of stuff because you have to assume a lot about the user.

**Mark Miller:** It changes the architecture from the beginning.

**Rory Blyth:** I was wondering about that; I mean, how do you do this without bringing the whole system to a grinding halt because it seems like you have to be constantly monitoring what's going on and keeping a lot of information.

**Mark Miller:** I know, performance in general is not that big of a deal. I think that there's probably -- I think to some degree there's memory usage, but I think even that you can work out to be pretty efficient. You can say well, how many times have they used this feature, you know, what's the time between availability and the time they use the feature. In other words, because I think you get a good sense of -- if I have got something available and it takes the customer 2 to 5 seconds to hit the hot key versus it takes them a quarter of a second -- you know, those two distinctions, I think you can make some conclusions based on that, and if you just track a few things on that -- a little bit of data, I think you can keep memory usage pretty efficient. There have been -- one of the things though that we've had to battle internally with context is there have been some cases where context does take a while to resolve; and its frustrating in those situations because you usually have to go to like a background thread if you are going to be really efficient, or simply maybe not make something available for a while, while you are in that background thread -- you know, trying to figure something out. A good example is, optimize name space references -- like in your imports or using statements; to find out which ones are not used and redundant, you have to resolve every single type in the source code, and if your file is big, that can take a couple of seconds to do that, and so, there may be a delay for a couple of seconds before we find out if we are available or not. And what do we do during that time, where we don't want to show an hour glass, we want to let the user work? So, you know, but there are challenges like that that are a little bit frustrating for me personally because its like, God, you want to have that information instantly, and what's the quickest time that we can get it? It's not quite as quick as you want it to be; and that's with today's machines.

**Rory Blyth:** So, what do you do if you aren't showing the hourglass? I mean, do you just show



nothing or do you say like, the documentation is being built or what do you say?

**Mark Miller:** You know right now, we don't show anything. But I really like the way that IntelliSense says, "Hey you know the background cache is being built, just hold your horses, it will be here in a minute." But we don't show anything now. Right now the delay -- I'll give you a good example of where this is currently effecting at least in the bills that I have got internally; this is for Refactor, and this is one of the new Refactorings that's coming out -- in this latest release of Refactor was coming out, like this week, or the week we are recording this -- is a cross file rename of members. But if you have -- like we've got a large project, it's got about 1000 files in it I think, and each of those classes is easily got, you know, several hundred lines of code, some up to like, 4000 lines of code -- I think it's the largest class...

**Rory Blyth:** It's going to take a while, yeah.

**Mark Miller:** And it takes about 20 seconds after you start up and you load for the first time before that rename Refactoring is available. So the other ones are all available, but the others -- that rename Refactoring in particular is not available, does not show up on the menu for about 20 seconds; you can see it work right, because its all happening in a background thread but its not available -- and I am struggling with how exactly are we going to solve this for somebody who opens up a project and wants it immediately renamed.

**Rory Blyth:** Right.

**Mark Miller:** And I have got kind of an okay answer, but I don't have a perfect answer.

**Rory Blyth:** Yeah, but a project of that size though; I mean, that's not average -- you know, that's a pretty large project what you are talking about that.

**Mark Miller:** I would say that probably about 10-20% of all .Net Visual Studio developers have got projects that are that big. I would say a good chunk of people are working in big projects -- or even bigger. Some of the -- you know, one of the things that I like to do is, I like to connect to really large files; it's kind of my hobby. So if I -- if I talk to a customer and they are saying, well you know, I have got a perf issue or something like this, I say, "Can you send me that file?" So you know -- and we also have -- we have some that generates files as well for us that are just very-very source code files, so we can do stress testing.

**Rory Blyth:** Interesting; okay.

**Carl Franklin:** Along those lines Mark, as far as discoverability and other products that you have, do you have any ideas for any of your other tools or is this, you know, this side window, this training window going to be a standard feature of all of your tools, does it only apply to CodeRush, what's the story there?

**Mark Miller:** You know what? Let me tell you, I want to -- let me divert off track and I will answer that question in just one second, because during a question I -- something occurred to me that was important that I want to say. With regards to discoverability -- and there is one other thing that we did that was very successful; and that was, we put a "Training Videos" menu item under our main menu. So, when they go in the DevExpress menu, like the fourth item down is "Training Videos"; they choose that item, it opens up a URL right inside Visual Studio and they can now start running the training videos and see the ones that we've created. I was talking to Scott Guthrie who works over with the ASP.NET guys in Visual Studio.

**Rory Blyth:** Yeah, we have heard of him.

**Rory Blyth:** I don't know if you know this guy or not; and anyway, he was telling me about all the training videos that they had, and I said, you know, I would strongly suggest you put a menu item in the product for the next version so that it says -- so that they can do the training -- so that customers can go in and just get right to the page if they want to.

**Carl Franklin:** Right, right.

**Mark Miller:** So, with that digression -- I feel like I am on the verge of one more digression before I answer your question, but you know, let me actually...

**Carl Franklin:** Let me respond to that while you are thinking of this...

**Mark Miller:** All right, go ahead.

**Carl Franklin:** Because you know, this -- video help and training videos is another good thing, but it can be very frustrating especially if its not quite what you are looking for -- you know, you have to sort of, be very positive that you are getting the right video, and especially if you are...

**Mark Miller:** I think granularity can solve that problem Carl; like one of the things that we try to do is, we try to do keep the videos focused or just show one thing, and keep them down to like 5 minutes -- or you know, if we can get a training

video to like, 5 or 6 minutes, that's like great for us; that's a nice little small chunk that somebody is willing to watch. So, we have got a lot of good feedback about those -- and I was talking to Scott and he was saying the same thing with their training videos they had a lot of great feedback but people got to go, you know, where the site is right?

**Rory Blyth:** Right.

**Mark Miller:** They've got to go out and find it, and again, you have got this discoverability issue and they could solve it with a single menu item; so, I really hope to see that. Now one more digression, because you asked me earlier how closely do we work with the Visual Studio integration guys and I didn't really answer that question...

**Rory Blyth:** No, you really didn't, Miller.

**Mark Miller:** Yeah I have been kind of -- and I just realized that, and the answer is, it varies -- sometimes very-very closely, there have been times when we have been in there with, you know, the people that are directly working on the editor system; where we have asked them questions about something and they have done the research, they come right back and give us the answer right there, so we are right there on site with them as part of our membership in the Visual Studio Integration program.

We have the ability to go up into ten special labs that they have up there, specifically for this purpose. So, sometimes it's very closely, and sometimes we just bypass it because there is nothing there, we know there is nothing there, and we just bypass it. So, that's the answer to that question. Finally Carl, you asked me -- now I've digressed twice from your question; I want to get back to answering it and you said -- and I think I have almost completely forgotten; it was something about other products and we are looking for in terms of those directions where we are going with discoverability. Well, the preview hinting for Refactor is huge. We are going to exploit the heck out of that and go forward in that direction. The training window is also very good too; I think though, that what's going to happen is, that's going to probably stay a CodeRush-only feature. And we might show the Refactorings that are available in that; we might create a separate tour window for that, I am not 100% sure.

**Carl Franklin:** Yeah, because I guess it sort of takes the place of IntelliSense, because CodeRush isn't really something where you are programming against objects, you are really hitting keystroke macros. So, you need something IntelliSense like, it doesn't quite fit.

**Mark Miller:** Right. I have got some objections to some of the ways that IntelliSense works I mean, you got to realize, it's a very challenging problem to solve. You want to give people a list of options, you want to get it close to where their eyes are looking, right, so you want to put it up where caret is in the editor, you know, the flashing cursor, the caret, you want to get it up there, but you don't want to hide code at the same time. So, you have got all of these conflicting constraints and it's really hard to achieve a good balance in there. So, but the thing is, I knew that we could not do it with IntelliSense; we couldn't solve it with a pop up window, it had to be dedicated on this side; I knew because of the amount of data that we had, there was potential to show there, and we didn't want to hide code. And the other thing is, because its discoverability, we wanted to allow people if they want to see it, they just look, and if they don't they look away -- I mean, they don't even look at it.

**Carl Franklin:** And I think I know the answer to this one, but what do you think of, like semi-transparent windows and things so you can sort of see what's underneath. I mean, that turns the screen into more confusing mush.

**Mark Miller:** You know what, I want to say one thing on this, because for those out there who listened to my last appearance here, which was I think "Mark Miller Wraps It Up," I accosted, which -- they've got a word accosted -- I think it is.

**Rory Blyth:** It is; you are probably it wrong, but go ahead.

**Mark Miller:** It's never stopped me before Rory; I just totally ripped Richard Campbell a new one, because of his passion for semi-transparent windows -- and, but the thing was, I was being pretty close-minded during that bit. I was not; I did not even bother to look.

**Rory Blyth:** Is this a long way of saying, "I am sorry I was wrong?"

**Mark Miller:** No, I was not wrong, and I am not sorry.

**Rory Blyth:** Well, you used accosted wrong.

**Mark Miller:** I do have a little pang of regret.

**Carl Franklin:** Okay.

**Rory Blyth:** Okay, a little pang of regret.

**Mark Miller:** Yes, the regret was -- my mind was not open to seeing what is the way he was using it, but in general, I find that there is a lot of abuse



when a new UI ability comes up, like fonts, when that first came out. I used to get these letters from my mom that looked at first like she was being held captive. You know?

**Carl Franklin:** Turn off the purple background.

**Mark Miller:** And so, like with Avalon coming out -- oh! By the way, you know, this is another great opportunity, because I want to talk about acronyms; just a couple of rules. If you're going to create an acronym, it is not allowed to use the word -- the letter 'W' in the acronym that has three syllables.

**Rory Blyth:** 'W.'

**Mark Miller:** Okay. 'W' -- it doesn't count, it doesn't work. So like, you got to throw out WPF and WCF.

**Carl Franklin:** Windows starts with 'W'; how can we do that?

**Mark Miller:** Well, I know, and then...

**Carl Franklin:** You need a new word for Windows; that's what we need Rory.

**Rory Blyth:** Well, you know what I have been trying to say Mark, as an Evangelist for Microsoft?

**Mark Miller:** Yes.

**Rory Blyth:** That's great feedback and I am going to take that back to the product team.

(Laughter)

**Carl Franklin:** So, another thing that you've brought up that you might want to talk about is approaching the achievement of perfection. What you mean by that?

**Mark Miller:** Yeah I was -- well, you know, it's occurred to me Carl, I was looking at one of our features a couple days ago. And I was thinking to myself, this is close to perfect. And for me it was a feeling of, like wonderful satisfaction and almost excitement every time I used the feature; every time I went in there I was like, yes, this is exactly what I want; show me just the information I need, and it's showing me in a way that's totally clear -- that kind of perfection is what I am talking about. I think by far 99 % of our product is far from perfect; I think it's good enough -- it's certainly good enough to sell and ship and to use as a customer, but I am really aware of just about everything I write, I am aware that in five years or in ten years, I'll figure out a better way to make it -- you know, polish it or perfect.

**Rory Blyth:** Yeah, right.

**Mark Miller:** And so, what I am really trying to do is, I am trying to create that thing that doesn't need any changes for like 20 years; and I found it to be extremely elusive. It is extremely challenging to create...

**Carl Franklin:** Let's start with the operating system, how about that man?

**Mark Miller:** Well, I am just working in my own domain; and I just -- I guess the thing is that it is - I really, really want it badly; I want it badly as a user I want it, because I use my own stuff. I think I had one guy say to me -- because, you know, when I use CodeRush, I work really fast -- I had one guy say to me, he said, Mark, you are the only guy I know that writes code blazingly fast and is extremely frustrated all the while that you are writing it -- and it's really, really true...

**Carl Franklin:** So, do I see this as an asset or a liability, I mean, is this a problem?

**Mark Miller:** To me?

**Carl Franklin:** Yeah.

**Mark Miller:** Well it's I guess....

**Carl Franklin:** Is this a character flaw maybe or?

**Mark Miller:** Yeah, I did leave my family at a bus stop once because of this. I mean, it sounds like something you got to a shrink for.

**Carl Franklin:** This portion of .NET Rocks brought you by telerik r.a.d. controls - the most comprehensive suite of components for Windows Forms and ASP.Net Web applications online at [www.telerik.com](http://www.telerik.com).

(Music)

**Rory Blyth:** It just -- it seems, I mean, no offense here; Miller, I love you to death you know that but it just -- I don't really love you. Well anyway, so, it's just seems kind of complicated because one of the issues about perfection right, is that its kind of contextual; I mean as needs change and system around it changes, suddenly that perfect thing becomes imperfect right, so...

**Mark Miller:** I know.

**Rory Blyth:** I mean, you Mark Miller, would have to like take over the entire tech-world and like, put a freeze on all future changes.

**Carl Franklin:** Or just be able to see the future.



**Rory Blyth:** Now, that's a good idea too, that's cool,

**Carl Franklin:** And I mean that sincerely, I mean, if you have enough vision.

**Mark Miller:** I do see the future and the frustrating thing for me is that we are -- some of the things that I see are just too hard to write right now, they take too much manpower and I mean, I think for the last -- I want to say for the last eight years, I have spent my life thinking, "I can't wait till next year when this feature is done" -- you know, in other words I know what's coming in the next year, and I have been -- can't wait, and each year something new comes in to fill in what's going to happen in the next year.

**Carl Franklin:** Mark, doesn't the pursuit of perfection in software like you're going for, doesn't that like, fly in the face of business practices, I mean...

**Mark Miller:** Well, you have to be...

**Carl Franklin:** Wait a minute; isn't the whole idea to have planned obsolescence so that you can get people to upgrade, I mean, if you write it perfect out of the gate, then what have you got? Especially if you sell it to China or something, then you're completely screwed.

**Mark Miller:** If you do a subscription model then - - then you are -- well, then you can -- well, I don't know, I guess to some degree, yes, you're right.

**Carl Franklin:** If it's perfect, why would I subscribe? I would just buy it once and be done.

**Mark Miller:** Because new features are coming out. Well and the other thing though is, the truth that -- and I should point this out so people don't think I am too much of a freak. (laughing)

**Rory Blyth:** Too late; no, go ahead though; nobody thinks that.

**Mark Miller:** The word perfection in -- the vocabulary word perfection really only came in to my thinking in the last couple of weeks. And it was mainly in the experience of this one small little -- part of this feature, where there's like, my God, that's it -- you know, that just feels so right - - and it's not like you spin this thing where I have been saying, okay here's where my threshold is, and I've got to meet it now

**Carl Franklin:** Sure.

**Mark Miller:** I mean, you know, from a practical business sense you have to release -- you have

to -- I think what makes sense to some degree or I think what's good is, if you have the goal of -- you know, this super high goal -- and that's your threshold for what you are shooting for; but then you have this other threshold that you say, this is good enough to ship; you know, this is good enough for high customer satisfaction right now to satisfy that -- but you are shooting for the top one, but as soon as you get above that lower threshold, you say okay, we can ship it now.

**Carl Franklin:** Yeah

**Mark Miller:** You know, we could satisfy it.

**Rory Blyth:** It's like dating.

**Mark Miller:** It's dating, yeah.

**Rory Blyth:** Good enough.

**Mark Miller:** That's what -- I agree with Rory I -- this is all I got my wife I said, "Honey you're going to have to lower your expectations."

**Rory Blyth:** Oh God! Good enough yeah.

**Mark Miller:** Your threshold is too high.

**Carl Franklin:** It is the problem; right here.

**Mark Miller:** Yeah.

**Rory Blyth:** Yeah.

**Carl Franklin:** Well, are you going to be speaking at any user groups coming up here in the near future?

**Mark Miller:** Yeah, actually by the time we broadcast it, I've spoken to one a week ago, I'm actually speaking to one tonight, the LA C# User Group and then I'm...

**Carl Franklin:** Okay. About the future though?

**Mark Miller:** In October I am going to something in Tennessee? I think, it's called devLink 2006.

**Carl Franklin:** Oh, yes, I'll there be too.

**Mark Miller:** You know, I should have had my schedule up in front of me, I just don't -- and I swear I'm doing something else some other time too and...

**Carl Franklin:** Are you going to TechEd Europe, Barcelona?

**Mark Miller:** I don't think I will -- I think there is something here at DevConnections at the same time, isn't there?



**Carl Franklin:** Yeah. There is TechEd in Barcelona and DevConnections in Las Vegas at the same time.

**Mark Miller:** So, I am pretty sure I am going to be at DevConnections and I don't remember why; I don't think I am speaking there, but there is something else important going on, and I just don't remember what it is; but that's, you know, come around.

**Carl Franklin:** Maybe it was the dancing girls at maybe -- no?

**Mark Miller:** No dancing girls, no. Ninjas maybe; I always wanted Ninjas to come down from the ceiling during my demo.

**Carl Franklin:** I almost hesitate to ask; have you seen anything online that very cool lately or, I mean I know you don't do a lot of surfing for tools, I mean you have a anti-Hanselman in that sense, but have you seen anything cool that strikes your fancy -- a toy maybe, a video game, I don't know.

**Mark Miller:** I got Startup Cop from PCs -- I think it is PC Tools -- and I was pretty pleased with it. It -- you know, here is a thing called -- I mean, you probably feel this way as well; you're a software developer, you load up some software and you are like -- ah, this could be a little bit better.

**Rory Blyth:** Yeah.

**Mark Miller:** You know, you want to change it, but you don't have source code.

**Carl Franklin:** Sure, I could fix it, yeah.

**Mark Miller:** So, you know, I got Startup Cop and it's not bad though. I like the results of that, so that would go in there -- and you know what, I want to --

**Rory Blyth:** Well, what is it though, can you tell us about it? I mean we know its name but...

**Mark Miller:** no, I am not going to tell you.

**Rory Blyth:** Well no, but tell me!

**Mark Miller:** Will everyone listen to an interview show?

**Rory Blyth:** Dude, you're a jerk. Just tell me; and you don't know what accost means, now tell me.

**Mark Miller:** I know what a cost means, that's like, how much you pay for something.

**Rory Blyth:** Mark Miller is a big jerk. Now tell me -- yeah, accost me with a description of what this product is?

**Mark Miller:** Startup Cop does, is it allows you to decide what's going to be starting up and what's not. So, you know you...

**Rory Blyth:** Like what Operating System or what?

**Mark Miller:** No, no I am sorry. What, you know like, you can have these things that load up into your -- you know, I want this to run every time and all this stuff like, you know, I think I downloaded iTunes player and then they got this thing that says, "Eh, I am going to startup every time this background stuff -- and I just wanted to clean up a lot of the background stuff, and I actually improved my startup time significantly because I was having some conflict between ZoneAlarm and some other stuff and..."

**Carl Franklin:** There ought to be a law against putting shit in the system tray, don't you wish , you think...

**Mark Miller:** Well, without telling you -- well, the system tray but also just the startup -- just you know, hooking yourself up to startup every time, McAfee is just an egregious violator of this. They start up, like I want to say, 4 to 5 different apps, just all running in the background and I am just like -- it makes me wonder. Did they do that just to make it harder for a virus to shut him down or is there something else going on?

**Carl Franklin:** Is there something different about this from just using msconfig and deselecting items that have been sent to the startup?

**Mark Miller:** Well, you know, I am kind of ignorant in terms of all the different things that are available.

**Rory Blyth:** Does it actually tell you exactly what application each of these startup features belongs to? Because if it does, then in that respect it would be better than...

**Mark Miller:** Yeah. It does have some of that information -- yeah, it's got a company name, so it shows you that, it has got a couple of options. Another thing that's cool about it Rory, is that you can set things up to startup after a delay. So you can say "Hey, after..."

**Rory Blyth:** That's cool.

**Mark Miller:** ...after I've been running for about 5 minutes now, I want you to start this up.



**Carl Franklin:** Because of course, the problem is that everything starts at once and your system just chokes for...

**Rory Blyth:** It's like 100 people trying to get in the same subway car at once, yeah...

**Mark Miller:** Right. And so, that's nice; you can also say things I want this to only startup when the network connection becomes available; that sort of thing.

**Carl Franklin:** That's good.

**Mark Miller:** So that's nice. The other thing that I wanted to plug if I could -- actually let me -- two things; one is, I'd love for people to take a look at the videos that we made for -- discoverability videos that we made for CodeRush, just to take a look at those things that are out there, and you can get to those at [www.devexpress.com/coderushtraining](http://www.devexpress.com/coderushtraining) and you can see those training videos and then the other thing I wanted to say is, I just want to kind of announce this is that we are wrapping up -- my team is wrapping up a support for C++ Refactoring. We will have a really strong beta of this available very soon and it's really, really significant stuff. I think it's -- C++ is a very challenging language to Refactor. It's mind-blowingly crappy.

**Carl Franklin:** To cut, to read, to refer to? Or what.

**Mark Miller:** Yes. I mean, this is almost topic for another discussion, but languages, are they written for -- you know, C++ is a language that was not written for a great design time experience; and in my opinion that's the future of languages; that's where I think languages are going to go, so that your design time experience in the IDE is going to be great; and C++ was not written with that regard.

**Carl Franklin:** Definitely.

**Mark Miller:** It is extremely painful to get all this right, and I am very proud of the work that our team has done on the C++ refactoring stuff. So, if you are a C++ developer, watch for it; I guess is what I was saying here.

**Carl Franklin:** Now CodeRush 2.0, is it out yet?

**Mark Miller:** Yeah.

**Carl Franklin:** Okay, and when did it come out?

**Mark Miller:** About, I want to say 3 weeks ago -- 2 or 3 weeks ago.

**Carl Franklin:** And just real quick if you haven't done it, what's new and cool? Real quick.

**Mark Miller:** Coolest thing are dynamic templates -- the coolest thing. So the idea is, if you want to create a new # table you type an NH and the spacebar -- and N is for creating new things and H is for the # table. And you can set up mnemonics for all the types and we ship with almost 80 mnemonics built-in for these types, but you can add your own with a simple mouse click. So, you can right click on your own type and you add a new mnemonic, and all the verbs, like N for creating new instances or M for creating Methods or P for Properties, all the verbs will now work with your mnemonic. And we also have mnemonics for generics as well; so, what happens is, if you want to create like, for example a -- say you've added a mnemonic "E" for employee, you've got a Class called "Employee," you want to create a new list of employees. There is a sequence of about 4 characters you can press to create a new instance of that that's initialized to type employee -- and the training window shows you each of these keys as you press it and also will show you your employee Class in that list -- and says, press "E" if you want employee -- if you want to add a new employee.

**Carl Franklin:** Oh, that's great.

**Mark Miller:** So it's really, really powerful, and I know you said real quick, but there is kind of interesting observation I have, I can't shut up on this. One of the things that we are doing with CodeRush is, we're really -- this occurred to me in the -- like the last couple of weeks, we are essentially in the business of compressing code; we are in the business of at least observing ways that code can be compressed. And so, if there is a short way to get the same thing a very, very short way, then by all means that's going to be part of our product. And what we are hitting up against just now, Carl, I am calling this "The Wall" and it's the level at which how efficient you can get before, you know, you can't get any more efficient than that and it's essentially due to limits of the keyboard; I cannot get anymore efficient than one key. There is no such thing as a half-a-key; I can't press half-a-key yet. And so, I am really waiting for that brainwave control.

**Carl Franklin:** Now you got to have one key that does like, 9 million things and it does them all correctly based on your...

**Rory Blyth:** You'll just be bitching later on, though Mark, you would be like, now I don't have a brainwave, and I don't know how to do it.

**Mark Miller:** That's what I need; the link I wanted to -- I wanted to give you this link because it's for



the brainwave -- let me give you this link because I think you guys will love this. It's [www.shrinkster.com/gk9](http://www.shrinkster.com/gk9). All right, if you go to this, the headline is, "Monkeys control robotic arm with brain implants."

**Carl Franklin:** I knew it would come down to monkeys.

**Mark Miller:** Okay. And this is the stuff that they've been working on at Duke University for a while.

**Rory Blyth:** How many monkeys do you need to control one arm?

**Mark Miller:** They only need one, only one monkey.

**Rory Blyth:** Like one monkey does a finger and other monkey does a finger? Do they work together; is the arm fully under control?

**Mark Miller:** No. It's a brain implant, Rory, it's a brain implant.

**Carl Franklin:** What if you had an infinite number of them, would they produce the complete works of Shakespeare?

**Mark Miller:** You are mocking me.

**Rory Blyth:** Well, yeah, that one was a little mocking, sorry.

**Mark Miller:** Alright. And no, they will not, by the way. I had an argument with my...

**Rory Blyth:** You don't have to accost Carl here. You don't have to accost Carl.

**Carl Franklin:** I deserved it. It's okay.

**Mark Miller:** I had an argument with our CTO who has a Mathematics Major background, and I told him, you know, math, it's just a subset of physics -- that's for people who can't handle the units, you know. So, he got a little irritated with me on that. Now, I was also telling him that, you know, there is no-- that infinity was just a really large number outside the domain. He got pissed off at that one too; so anyway, but what was I saying, Oh, the monkey thing, they got chips that you can put on your brain and they can essentially read your mind just like you would do speech recognition -- in other words, they see all the patterns and they can say, "Hey you know what, he is moving his arm up and down, that's what these patterns are, okay, so now you can think it and you don't have to move your arm up and down, but we still recognize it."

**Carl Franklin:** One thing I want to leave us with is, that we have thought about and talked about doing a series on dnrTV where we'd called "Behind the Products" or something like that where we would actually have you talking about some of the techniques and the technologies that you use to create your tools, and not just CodeRush and Refactor because we sort of, talked about them a little bit, but you know, some of the other things that you've done; Developer Express has a lot of tools -- and so, "behind the tools" I think you would call it -- yeah, and we were going to do the same thing with DataDynamics and telerik and\_ just have them uncover some of their Source Code and some of their techniques and maybe even share a little bit with us. So, I think that would be a good idea, don't you?

**Carl Franklin:** I think it's got potential to be exciting man.

**Mark Miller:** Yeah, could be; and Rory, you have something you want to plug?

**Rory Blyth:** I'm plugging -- I want to plug, yeah. I want to plug my team, which is the Microsoft MSDN Events Team. We go around the country to all 50 states and we give free presentations to software developers and sometimes we even give away things like food and prizes and the URL to that is [www.msdn.events.com/rory](http://www.msdn.events.com/rory). Okay. It's free all year round; pretty cool, so check it out.

**Carl Franklin:** You've done some of your own screen casts too on mobility stuff little tiny things or...

**Rory Blyth:** I actually do a lot of things -- like, I have a lot of peripheral sort of activities I am involved with, so I do a lot of user group talks and I do code camp and all that, and recently, I did a lot of -- I did a bunch of Windows mobility screen casts upon channel 9. So, if you got to channel 9 and do a search for the windows mobile tag and then go to screencast, you will find a series of screencasts where I take you from start to finish over a series of screencasts, each one of which is no longer than 20 minutes, because I don't want to waste your time and I try to break things down to the proper level of granularity so that you just learn like, one particular feature at a time starting with how to build a mobile app, all the way to how to deal with SQL Server mobile and remote data access so that you can synchronize between SQL mobile and SQL server instance -- and I'm going to be putting more of those up before too long.

**Carl Franklin:** Sweet! And we'll put links to those on the .NET Rocks! page for this show. So, in



case you didn't get that URL, just go to the show page and -- from where you got this file, [dotnetrocks.com](http://dotnetrocks.com) and it would be there. Well guys, that's a wrap.

**Rory Blyth:** It was awesome.

**Carl Franklin:** Thank you Rory for stepping in.

**Rory Blyth:** Thanks for letting me step in.

**Carl Franklin:** Thank you Mark for sharing your brilliance.

**Rory Blyth:** And for accosting us.

**Carl Franklin:** And on that note we will see you next week on .NET Rocks!

**Geoff Maciolek:** .NET Rocks! can be found online at [www.dotnetrocks.com](http://www.dotnetrocks.com) and at [msdn.microsoft.com/dotnetrocks](http://msdn.microsoft.com/dotnetrocks). .NET Rocks! is edited each week by Geoff Maciolek, that's me, and Carl Franklin, who is also Executive Producer. All music heard on .NET Rocks! including ToyBoy, the theme song, is created and produced by Carl Franklin and Franklin Brothers Band. Carl never sleeps. .NET Rocks! is produced for franklins.net by PWOP productions - providing professional audio and podcasting services online at [www.pwop.com](http://www.pwop.com). PWOP - it's time to get your impact back.