



[HTTP://www.dotnetrocks.com](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 #483
(Transcription services provided by [PWOP Productions](#))



Kent Tegels on SQL Server Integration Services September 22, 2009

Our Sponsors



Developer
EXPRESS

[HTTP://www.devexpress.com](http://www.devexpress.com)



CoDe

component developer magazine

[HTTP://www.code-magazine.com](http://www.code-magazine.com)



[HTTP://www.telerik.com/](http://www.telerik.com/)



Kent Tegels on SQL Server Integration Services September 22, 2009

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]

Lawrence Ryan: Hey, Rock heads! Stop massaging your dirty bits 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 #483, with guest Kent Tegels, recorded live Tuesday, August 25, 2009. .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 Red Gate Software, the essential tools for SQL Server, .NET, and Exchange. Support is also provided by CoDe Magazine, the leading independent magazine for .NET developers, online at www.code-magazine.com. And now, the man who says, "Healthcare bill? I got a mailbox full of 'em," Carl Franklin.

Carl Franklin: Thank you very much and welcome back to .NET Rocks! You know, Lawrence always comes up with good jokes, doesn't he?

Richard Campbell: He does indeed.

Carl Franklin: Well, sometimes.

Kent Tegels: Yeah, but then why doesn't he ever tell them?

Carl Franklin: I don't know. Hey, you're not supposed to talk yet. Get back!

Richard Campbell: Control your guest. What's up with that guest?

Kent Tegels: It's a talk engine, man. It's talk engine.

Richard Campbell: It's an unruly guest.

Carl Franklin: So, this is how it's going to start.

Richard Campbell: This is how the show is going to go, man. I'm telling you.

Kent Tegels: Just call me the Jim Harrison of .NET Rocks!

Carl Franklin: All right. Well, let's get right into Better Know a Framework then.

[Music]

Richard Campbell: All right.

Carl Franklin: Better Know a Framework, of course, is a little segment I do where I just pick out a little piece of the .NET Framework that if it sounds interesting to you you can go look it up in the documentation. It's not training so don't send me emails. So today I'm talking about the IntPtr Structure.

Richard Campbell: Oh.

Carl Franklin: If you've done any P/Invoke or Windows API calls or standard Windows DLLs through P/Invoke. You know that IntPtr is a platform specific type that's use to represent a pointer or a handle, and it's IntPtr because that's usually an integer but it's not necessarily always 32-bit, or 16-bit, or 64-bit. It's designed to be an integer whose size is platform specific, that is an instance of this type is expected to be 32-bit on 32-bit hardware or operating systems, and 64 bits on 64-bit OS's.

Richard Campbell: This is part of that whole run any CPU mode when you build your app.

Carl Franklin: Yeah, which is evil.

Richard Campbell: I'm with you.

Carl Franklin: There are so many problems with using any CPU. If you go to the compile options in Visual Studio, probably a good idea is if you're not using extremely large data structures, extremely large files, that kind of things, if you're not using gobs and gobs of RAM you should just compile for x86 because it's going to run better even on 64 bit systems.

Richard Campbell: I totally agree and I think the only reason we're running into these issues these days I'm certainly finding in an ASP.NET is that people are now running 64-bit operating systems and finding out the hard way that actually their app doesn't run 64-bit properly.

Carl Franklin: Right and besides the benefit, and we said this before, the benefit of a 64-bit OS is more RAM but that means more 32-bit processes can fit into your large RAM pool.

Richard Campbell: Right.



Kent Tegels on SQL Server Integration Services September 22, 2009

Carl Franklin: I don't know how else to say...

Richard Campbell: What does that have anything to do with IntPtr?

Carl Franklin: I don't know, we just get off that, get off on the tangent somewhere. But the other thing that I want to mention about IntPtr is that you'll probably run into this if you're doing P/Invoke and a great resource for all of those declare statements is pinvoke.net.

Richard Campbell: Nice.

Carl Franklin: There it is.

Richard Campbell: There you go.

Kent Tegels: Exciting.

Carl Franklin: What have you got, Richard?

Richard Campbell: I've got an email and it goes back a little ways from about show 445 which was, you know, 40 shows ago. "Hi guys. I just want to drop a note to say thanks for another great show. I appreciated the discussion on REST, WCF, and MVC. In particular, I like the idea of building REST services on top of the ASP.NET MVC stack. I've recently been thinking along the same lines. In fact, using MVC to serve the basic JSON REST request was sort of a self-justification for my diving into the MVC framework in the first place. I felt that if I worked on learning MVC, I would have the added benefit of being able to build simple RESTful services. There are so many technologies that a person can try and wrap their brain around. There are only so many hours in a day I can commit to learning WCF and the many additional benefits that it brings to the table REST and otherwise at a later time." So his real point here is this whole get into MVC to get the advantages of REST without having to deal with WCF.

Carl Franklin: Yeah.

Richard Campbell: "Thanks for the great show and I'm not totally off-base in this case anyway. Tim Drowd from Anaheim, Saskatchewan." That's right, slipped the Canadian in there.

Carl Franklin: Saskatchewan, isn't that in the middle of nowhere?

Richard Campbell: Saskatchewan is one of the places where you want your dog to run away for three days.

Carl Franklin: Kind of flat, huh.

Richard Campbell: Yeah, it just keeps going. Well, it depends. I don't know exactly where Anaheim, Saskatchewan is. There are some places where they also keep their mosquitoes and there are some lakes and things but not just plains. Anyway Tim, thanks for your email, a mug is on it your way to Saskatchewan, and if you've got questions, ideas, concerns, suggestions, send us an email, dotnetrocks@franklins.net.

Carl Franklin: That brings us to our guest who has spoken before here, just a few minutes ago. You heard his voice out there. Kent Tegels is a mentor with PluralSight and a professor of computer science at Colorado Technical University. He teaches classes in databases, mathematics and economics. Oh man, we're screwed, Richard. We've got an academic here. He is going to tear us into pieces.

Richard Campbell: This is a smart guy.

Kent Tegels: I promise to behave. Well, maybe.

Carl Franklin: Kent is a former Microsoft MVP for SQL Server and holds numerous Microsoft certifications. He is completing his doctorate in - oh, we're really screwed.

Richard Campbell: Nice.

Carl Franklin: He is completing his doctorate in computer science at the Institute for Advanced Studies. Kent is a well known regional speaker, has co-authored a number of books and articles on Microsoft's .NET and web platforms. He lives in Sioux Falls, South Dakota and enjoys cooking. You can follow him via Twitter and Facebook as "KTEGELS." Welcome, Kent.

Kent Tegels: Thank you guys. It's nice to be here.

Carl Franklin: North Dakota, oh.

Kent Tegels: Yeah, it's north of me.

Carl Franklin: Oh, you're in South Dakota.

Kent Tegels: I'm in South Dakota.

Carl Franklin: You're in South Dakota, so...

Kent Tegels: Yeah and granted, there's not a lot of difference between the two but...

Carl Franklin: Yeah, Fargo's is in North Dakota.



Kent Tegels on SQL Server Integration Services September 22, 2009

Kent Tegels: We like to maintain our distinct identity for some reason.

Carl Franklin: But Fargo is in North Dakota though.

Kent Tegels: Fargo is there on North Dakota.

Carl Franklin: Yeah. And Mount Rushmore, south?

Kent Tegels: It's in South Dakota.

Carl Franklin: South Dakota.

Kent Tegels: It's about 400 miles west.

Carl Franklin: Yeah. Okay, enough about that. Let's talk about SQL Server Integration Services.

Kent Tegels: I like SQL Server Integration Services. That's a good idea.

Carl Franklin: Yeah. So this is your thing.

Kent Tegels: It's one of my things. It's probably what I'm most known for. It's a very slick set of tools and technologies and I think it's something that's really unappreciated by the developer community so I'm happy to be here talking to you, the developers, about SSIS today.

Carl Franklin: So if you're a developer who occasionally works with databases and you don't really do much on the DBA side, just tell us why SSIS and what problem does it solve?

Kent Tegels: Well, do you ever have to put data into a database?

Carl Franklin: Occasionally.

Kent Tegels: Occasionally, right. I used to argue that almost every application that existed somehow work data -- need most of the time that it went into a database.

Carl Franklin: I like to say the ones that pay money needs a database.

Kent Tegels: Yeah, yeah. The ones that actually are not academic.

Carl Franklin: That's right.

Kent Tegels: Their job is basically here's some data, manipulate it, and somehow get it into the database.

Carl Franklin: So the idea is you have data from somewhere else, now you want to load that into SQL Server.

Kent Tegels: Right. You can even be generating it as a result of reading devices, you could be pulling it off of an alien system like an AS/400 or something like that, whatever it is. But most of the programs I've read in my life, they've been that where we have data in format X and we need to get to format Y and how do you most efficiently do that.

Carl Franklin: The older technology was Data Transformation Services, right? DTS?

Kent Tegels: It was good old DTS, you bet ya.

Carl Franklin: Yeah.

Kent Tegels: And DTS was an interesting tool. I'll leave it at that.

Richard Campbell: It was really a programmer's tool because you had to do anything that was of any interest at all in VBScript which it really hits the old wayback machine.

Kent Tegels: Yeah. I'm not sure that that makes it a programmer's tool but it does make it more developer-focused. That's for sure.

Richard Campbell: My point being most DBAs look at that one and say "I blame you" when they pointed to the developer.

Carl Franklin: Yeah.

Kent Tegels: Yeah, absolutely. Right.

Carl Franklin: Because there's script and all.

Kent Tegels: Because you can do pointers in VB so you can really point at the developer.

Carl Franklin: Ah.

Richard Campbell: Ah, I see. So yeah.

Carl Franklin: Now if you have VB.NET you'd use an IntPtr and be...

Kent Tegels: There you go. Those tips are always helpful, those little chunks of the .NET Framework.

Carl Franklin: Yeah.



Kent Tegels: We always find a good way to tie those into the show, don't we?

Carl Franklin: Absolutely. I think.

Kent Tegels: The big thing for me with SSIS is I got really tired of writing data access code. Why trouble to write – this will age me quite a bit, but why trouble to write SQL command adapters and why struggle with data adapter and all that.

Carl Franklin: Yeah, it just works.

Kent Tegels: Well, it had just gotten more and more hideous as time has gone on. We had the enterprise library for a while and boy, everybody thought that was going to be the answer, and well, we've kind of see where that's going for data access. Then we got LINQ, and well, I get the whole show on why I don't like LINQ but I'll stick to SQL.

Carl Franklin: You don't like LINQ.

Kent Tegels: I don't like LINQ. No, and I really, really don't like LINQ to XML but that's for...

Carl Franklin: Well, we've got to go down this rabbit hole. Why?

Kent Tegels: Perhaps because we already had something in the .NET Framework that was better.

Carl Franklin: Better?

Kent Tegels: Yeah, it's called XQuery and it was a...

Carl Franklin: Well, that's for databases though.

Kent Tegels: No, it's not that at all.

Carl Franklin: What if you just got a huge collection of data that you want to pull stuff out of and it's just in memory?

Kent Tegels: That's the one use case for LINQ that I think is very valid, it's LINQ to Collection.

Carl Franklin: Okay.

Kent Tegels: I have no problem with that one whatsoever, but again where should data live? Where is it going to be persistent? Where is it going to back up? Where is it going to secure? Is that going to be an application store? Probably not.

Carl Franklin: Well, there are times I would argue that where the data exists like in a file tree or

something like that, if you have -- let's say you're writing a program to automatically pull up your MP3 files and it has to go through directories and find all these MP3's, take the tag data, put it into classes, load it into a collection, it's not a lot of data but you may have 4,000 or 5,000 songs in there and it's going to change every time so you have to read it from the files themselves, and now you have to pull that into a collection and...

Kent Tegels: Oh, do you?

Carl Franklin: The way you query -- well, you don't have to but why would you put it in a database when it's going to change next time you run the program?

Kent Tegels: Because it may or may not change next time you run the program. It depends on how the velocity of which you add data to the collection, and also depends on exactly what you're doing with that data. You may want to make multiple query available to it which an application has a hard time doing, and it's funny that you mention that because one of the demos we normally do when I talk about this is Scripting is exactly that. How do we enumerate through a directory of MP3's, how do we load it into the data buffers, how do we ship that in for the metadata out to a database?

Carl Franklin: I would argue that up to a certain number of songs and sizes and hard disk speeds, it's just much faster to read and load that data right off the disk and it's more accurate and doesn't require a database at all.

Kent Tegels: I think the only thing that it doesn't require is it doesn't require ongoing maintenance of the data. Other than that, I would challenge you to write a LINQ Query that's faster than writing SQL.

Carl Franklin: So challenged. I have written it.

Kent Tegels: Yeah, at some time let's sit down and do that.

Carl Franklin: Yes, we will.

Kent Tegels: Yup.

Carl Franklin: Okay, so let's pop back up the rabbit hole and...

Richard Campbell: I just want to throw one other element in the LINQ which is you said XQuery, and I don't inherently hate XQuery, I just don't do it often enough to be good at it.



Kent Tegels: Right.

Richard Campbell: And it's one of those things where if you don't maintain that skill it bleaks away.

Kent Tegels: Oh, absolutely.

Carl Franklin: Not me, I hate it.

Richard Campbell: I mean I even got good at XSLT at one point in my life but it got away from me too.

Carl Franklin: I hate that too.

Kent Tegels: Yeah.

[Laughter]

Richard Campbell: Yeah. And I think one of the best strengths of LINQ is because it's a common querying language across all of these different things. It's more likely you're going to retain it.

Kent Tegels: Well, it's more likely you're going to retain it if you use it just like anything else. The more you use it, the more you retain it.

Richard Campbell: Yeah.

Kent Tegels: I studied LINQ for a long time and I can still write queries in it but what's my Query language of choice? Well, it's T-SQL.

Richard Campbell: Right.

Kent Tegels: So I think personal bias is obviously the plan to it. I don't harbor anything against the excellent people other than we already had a good way to query XML in the .NET Framework, and Microsoft put a lot of money into XQuery and they just kind of let it get away and I thought that was particularly sad.

Carl Franklin: This portion of .NET Rocks! is brought to you by our good friends at Telerik without whom this show would not exist. No doubt you bump into testing tasks now and then in your work and we can bet writing functional test is not your favorite thing. It's difficult. It takes ages and the results could be dubious. Well, get ready to start liking it, thanks to Telerik. With the just launched WebAii testing framework, building web automation tests is a breeze. Enjoy codebase automation of advance ASP.NET AJAX and Silverlight apps. Write a single test and have it executed against multiple browsers at once. Benefit from rich API LINQ support, integration Visual Studio unit testing NUnit, xUnit, and MbUnit, not to mention the free wrappers for a Telerik RadControl for ASP.NET AJAX and Silverlight as shipped with

Telerik's new testing tool. Surely one of its best features, WebAii testing framework which is developed by ArtOfTest, is absolutely free. If you're already hooked on WebAii testing framework, you can start using it right away. Go to www.telerik.com for more info. And hey, make sure you thank them for supporting .NET Rocks!

All right, let's get back to SSIS.

Kent Tegels: Okay. So I think developers really need to know about it because it doesn't make any sense to write the data plumbing. Why should you invest a lot of effort into writing the interfaces to read the data and to write the data?

Carl Franklin: Yeah.

Kent Tegels: SSIS does that for you and I think that's one of the big things. The other one is it's highly paralyzed internally and it does a great job of memory management. Those are particularly hard things for developers, particularly new developers, to really do very well. So from my point of view, I use SSIS for just about everything rather than write my own code.

Carl Franklin: When you say everything, are you talking about just when you have data that isn't in the database and needs to get in there?

Kent Tegels: Well, you know, that's one use of SSIS. Another use is to send the data to no place and that may sound like a really strange thing but one of the things I do is I do a lot of statistical analysis of data, and what I use SSIS for is I use that to read the data source, use some various SSIS components to clean it up and to get it in the shape I want, and then I use the built-in visualizers in there to see what the data is actually doing for me as a result of those processes.

Carl Franklin: Yeah.

Kent Tegels: So it's not exactly a one trick pony. It's just like anything else. The more you think outside the box, the more uses you find for it.

Carl Franklin: So what is it not useful for?

Kent Tegels: It's really not useful for the web. Yeah, it's useful as background stuff. It's really useful on the server-side. You're not going to produce videos with it. You're not going to make dancing curves appear on a page. That's not at all what it's intended to do. The visualizations are pretty rough right now. I can draw maps with it but their point maps so it's like an artist tipping out a map of the United States rather than having a nice raster version on the apps.



Kent Tegels on SQL Server Integration Services September 22, 2009

Carl Franklin: Well, now you're mentioning UIs, so there's UI involve in SSIS?

Kent Tegels: There is to a degree. It's fairly primitive at this point but you can use it to do some visualization, sure.

Carl Franklin: In a Windows Form application only?

Kent Tegels: No, inside the VS. Basically, you'd be running it inside the Visual Studio or Business Intelligence Development Studio and looking at the various what are called data viewers, and you look at the data viewers.

Carl Franklin: And the data viewers are for you as a developer, not for the end-user?

Kent Tegels: I've actually used them in academic presentations where we're looking at Scatter Plot and that's how I visualize the Scatter Plot.

Carl Franklin: I guess I'm just trying to figure out what this is. Is it custom control?

Kent Tegels: It's a control that's embedded inside of the runtime, the debugging runtime, for SSIS.

Carl Franklin: So it isn't something you plop on a Windows Forms.

Kent Tegels: No.

Carl Franklin: Okay.

Kent Tegels: It's not something you'd want to use in that sense.

Carl Franklin: Right.

Richard Campbell: So if you want to invoke an SSIS package, do you have to go to the SQL Server Management Studio for that?

Kent Tegels: No. You can just invoke it from the command line.

Richard Campbell: Okay.

Kent Tegels: Because there is a process called BTextact that all run those jobs for you. So realistically you could use SSIS as a background processing engine to take fairly complicated data extract out to simplify data you want for a web control or for a visual control and map that out.

Carl Franklin: I see. So a package that Richard had just talked about is the combination of a set of data and some queries that are going to run or some commands that are going to run to move that data from one place to another.

Kent Tegels: You bet ya.

Carl Franklin: Yeah.

Kent Tegels: For example, it's not all that atypical for me to do GEO special queries and get that data and visualize it hat way.

Carl Franklin: Now, can you -- obviously you can pull data out of a database with it and then do some, as you're saying, do some analysis on it, and then put it back into another set of tables like temporary tables that you can then run queries on? I guess I'm trying to figure out what other uses for it besides just analysis and moving of data like, you know, what practical uses do we have.

Kent Tegels: Okay. Have you ever had to take data and put it into a word document?

Carl Franklin: Oh yeah.

Kent Tegels: Do you ever had to take data and write it to an Excel?

Carl Franklin: Well, yeah.

Kent Tegels: Do you ever have to take data and try and a PowerPoint out of it?

Carl Franklin: I have indeed.

Kent Tegels: Okay. Well, SSIS is pretty good, or at least reasonably good at one of those three things. It's very good at generating Excel files. What some work -- you can get it to do word files pretty well. The last one, PowerPoint, that's something I've been doodling on about how can I take data out of a database and put it in the PowerPoint. For example if you're doing a command staff briefing for the military, they love to have PowerPoint and their PowerPoint decks are really death marches if you can pardon the pun. And boy, there are just all kinds of data in there and it goes on and on and on. I've been looking for ways to automate that for a long time, to automate the generation of PowerPoint decks. But if you got the data in a database and you got SSIS, it's sort of feasible now that we have open XML format but it's still a lot of programming work to be done.

Richard Campbell: I can't take this package and integrate it into my app? It sounds to me like you've got a great import/export engine here.



Kent Tegels on SQL Server Integration Services September 22, 2009

Kent Tegels: Well, you do but you don't get a DLL when you compile it. You get something that runs in an SSIS at runtime. So you do have to have a second runtime on there for the SSIS bit. So yeah, you can still show on vacation but that's how you would do it.

Richard Campbell: And what is this runtime? Like if I was going to distribute it to multiple machines, what do I got to ship?

Kent Tegels: You would have to ship SQL Server.

Richard Campbell: Which means I've got to ship licenses too, or do I get it with Express?

Kent Tegels: It's not in Express.

Richard Campbell: Ugh.

Kent Tegels: Yeah and at this point it doesn't look like there's going to be a mono-type version of it. There have not been a lot of people in the mono community that have put this inside of anything they want to invest their time into.

Carl Franklin: Well, is it because it's so closely tied to SQL Server?

Kent Tegels: I don't know that that's necessarily the case. I always felt that Microsoft sort of miss the boat on that one. Yeah, it works great with SQL Server but it's so far behind. Yeah, you can do so much more than just SQL Server work with it.

Carl Franklin: Sure.

Kent Tegels: But it really should, in a sense, be it's own product. Now, of course I'm not going to complain because when I pay SQL Server license, I get a license for it and that makes me pretty happy. I'd hate to pay the separate license fee but, you know, there are a number of products out there that do basically what SSIS does that they're happy to take your money. So kudos for Microsoft for keeping it free at least in the sense of it's included with SQL Server license.

Richard Campbell: But the limitation here from a developer's perspective is this is not something you roll and distribute with your app. It's something that might support your app but it's living on the SQL Server.

Kent Tegels: The package can live anywhere but you do have to have the runtime, and in order to have the license to the runtime you do have to have a license for SQL Server.

Richard Campbell: Right.

Kent Tegels: That's how I understand it.

Richard Campbell: I always thought that SSIS is real job. Like what it was built for was the ETL element of an OLAP cube.

Kent Tegels: I think that's definitely the primary reason it exists, but it's not the only reason to use it.

Carl Franklin: Okay. Give me some other practical uses.

Kent Tegels: Some other practical uses, let's see. There's definitely visualization, there's generation of XML if you need to do certain information out in an XML format to a specified schema. It's not very hard to do that with SSIS at all.

Carl Franklin: Have you seen XML Literals in VB.NET?

Kent Tegels: Well, there are two problems with that. It's XML Literals so it's VB.NET.

Carl Franklin: Ooh, ow.

Kent Tegels: Well, it's funny because I was listening to my good friend Don Demsak's show where he was talking about the Literals on the way over here and Don and I kind of -- we have a friendly rivalry about XLinQ and its relative value. I've been in the SQL Server community. At one point I guess I was known as Mr. XML. So it was Don XML in SQL for a while but all of that, the compiler magic and all of that, that's one way to solve the problem. Sometimes you just like to sit down with XML writer and get the job done.

Carl Franklin: Okay. Good luck with that. Right on with...

Kent Tegels: I've been doing it for years, Carl. It works just fine for me.

Carl Franklin: Good. I am an impartial talk show host.

Kent Tegels: Again I'm showing my age, I know, but I thought it actually was pretty cool too so...

Richard Campbell: You know, one of the things when I experiment with SSIS that I really appreciated, especially from a point of view of a guy who's writing his own parser for data, was its error handling capabilities.



Kent Tegels on SQL Server Integration Services September 22, 2009

Kent Tegels: Absolutely but pretty much for the most part we've gone away from writing our own "how do I crack this file open and work with it" because there are only DBC providers available now which makes using SSIS just essentially very, very easy to do.

Richard Campbell: Yeah, which is all well and fine right up until there's some kind of irregularity in the data.

Kent Tegels: Right.

Richard Campbell: That for me has always been the bare part of this where you ended up going down to the Bare Metal and write your own parser because there's an illegal character in it that make the thing barf and so you just have to read this binary and parse it yourself. Like all that stuff you have to do by your hand is really nasty.

Kent Tegels: Right and I don't want to give people the wrong impression that there's just automatic magic that makes that work, but it's very easy to redirect those rows to a rework file with SSIS and at least get your known good data into the system.

Carl Franklin: Now what's a rework file?

Kent Tegels: A rework file would be lines that have damaged data in them that are going to require some kind of manual or other processing on them. For example, you know you could have a directory full of MP3s, but ID3 tags they're not guaranteed to be there.

Carl Franklin: Right.

Kent Tegels: Since they're just, for the most part of it, they're just strings. If you're trying to convert them into something meaningful, sometimes you have to go through and figure out a different way to parse them each time.

Carl Franklin: Okay.

Kent Tegels: One of the very cool things, I think, about SSIS is its ability to do scripting in line to deal with some of that. Basically, my favorite class in the .NET Framework is `System.Text.RegularExpressions`.

Carl Franklin: Yeah.

Kent Tegels: I use the daylight's out of that for cases where, okay, the data is this, sometimes the data is this, sometimes but every Tuesday when there's a full moon and it's between April and March,

then you have to do this. You know, it makes that all bit more approachable.

Carl Franklin: Right.

Richard Campbell: Especially when you're dealing with, you know, I get this million row import and somewhere in the last 5% of this there's a really screwed up row and it makes the whole thing barf, does that mean I can add in a feature and just say, okay, when you can't handle that row, stick it over here and finish what you can finish?

Kent Tegels: Yeah.

Richard Campbell: That to me is brilliant.

Kent Tegels: Right.

Richard Campbell: So I end up with these 20 rows that are screwed up that at least I can go look at on their own.

Kent Tegels: Right.

Richard Campbell: But the rest got parsed.

Kent Tegels: Absolutely, absolutely and the rest, you know, usually applications are smart enough to realize that they may not have all the data and the data may not be complete, and yeah, there are some data that's going to be adjusted over time so I think a rework file works really well for that.

Carl Franklin: Okay.

Richard Campbell: Have we really talk about the sort of build experience of creating a package like this. What is it like?

Kent Tegels: It's Lego Programming.

Richard Campbell: Oh man.

Kent Tegels: It's just like many of the visual designers. I'm not a big WWF, or WSN, or whatever it's being called these days, the Workflow stuff, but I understand that that's a Lego design experience where you have a Surface, you drop controls on it and you wire the controls up. That's very much like what SSIS does. You have two different types of work. We have two different elements in a workflow. We have tasks and you have data flows and what you do is you start with the task, you drop whatever preparatory things you need to do onto to that, like maybe I need the FTP to get some files down. Or maybe I need to call a web service that gives me a bunch of discount rates for treasure notes or whatever it is. Get that data, do whatever preparation, you do Truncate Tables, do database consistency checks,



Kent Tegels on SQL Server Integration Services September 22, 2009

whatever it is you need to do, then you have a control on that form called the data flow and that's really where the action happens, that's really what causes -- it's really where you do more like Lego Programming and say, okay, here's my data source, here's my data destination. What do I have to do to transform that data in memory as I'm moving up through the system?

Carl Franklin: You're primarily using T-SQL or is there more to it?

Kent Tegels: You're really not writing any T-SQL at all.

Carl Franklin: Oh really?

Kent Tegels: Really all you're doing is you're snapping together, you're wiring up these Legos, they're .NET controls, they're just specific to the SSIS runtime, and you can insert a script control in there. In SQL Server 2005, I like to make the joke that you can program at any .NET that you'd like in SSIS as long as it was Visual Basic.NET. In 2008, we had a great improvement. You can write in any programming language you want as long as it's Visual Basic.NET or C#.

Richard Campbell: So the COBOL .NET guys are still waiting by the sidelines.

Kent Tegels: Unfortunately, those of us that like COBOL .NET are waiting by the sideline.

Richard Campbell: Both guys.

Kent Tegels: That's not entirely true though, because using scripting we can call a DLL that we've written in another language. So I could write or I couldn't, although I would probably want to eat a gun if I get this, because I could really write my very serious number of crunching or data processing logic in COBOL .NET, compile it, catalog it into the GAC, copy it to a specific directory where SSIS is going to look for it, and then write some script that calls that.

Richard Campbell: Right. Yeah, you always have that option. Once you bridge to anything .NET, you can bridge to your own assembly.

Kent Tegels: Absolutely right and Microsoft gives you your ability to write your own tasks and components too using the IDTs interface. So if you really have a great idea for DTS control you need, like I'd love to have a DTS control that does regular expressions for me and not a task and a component that does those for me, and I've looked into the feasibility of writing a component to do that so that I have a binary thing I can just ship to distributors and I don't have to have a lot of other stuff and it manifest

its own UI in the DTS design environment. So there's plenty of this, plenty of programming that can be done with the SSIS. If you really need extensibility, it's a great platform for it.

Richard Campbell: Now you start getting into this whole at what point am I crossing the reasonable line in an SSIS package?

Kent Tegels: Well, I guess I look at it from my point, it's there's no line of unreachability but other people may have a different opinion of that.

Carl Franklin: You're still waiting for the case in which you can't use it.

Kent Tegels: Pretty much. I'm still pretty much waiting for the case where I can use it.

Richard Campbell: Yeah. Couple of things, Kent, that I certainly run into DTS back in the battled days.

Kent Tegels: Uh-hmm.

Richard Campbell: Source control and backing these things up and migrating it from machine to machine.

Kent Tegels: Yup. TFS anybody?

Richard Campbell: What's that?

Kent Tegels: Anybody got -- are you running TFS now?

Richard Campbell: Aren't we all?

Kent Tegels: Well, no, not everybody but I think the majority of shops are doing that and SSIS is just another friendly member of the Visual Studio family that integrates quite nicely with that. Deployment can be a little bit of a sticky issue with SSIS. Depending on which control you use, you may inadvertently pin yourself to having to run a package on the local machine which is really not a very good idea.

Carl Franklin: No.

Kent Tegels: There is one destination adaptor that works with shared memory provider only and that means you have to run the package on the host that your writing the data to. Other than that, for the most part it's pretty straightforward because the file that it generates is just XML. When you save a package you design in the designer, it's just XML.

Richard Campbell: Nice.

Kent Tegels: So you could take it around at any place, and what's very, very nice about SSIS is

that it has a great configuration system that you can overwrite parts of that just by simply saying here's a fragment that I want you to apply over the top of it. It's called configuration and those can come from the database, they can come from an XML file, they can come from the environment, and they can come from the registry.

Richard Campbell: And what would be in this thing? The actual name of the database or name enable, that kind of thing?

Kent Tegels: Sure. Let's say that you have a connection string to destination database that you want to write to. Well, it doesn't make any sense to leave that hard coded in the package. No, probably not. What you do is you create a configuration and then you'd say, okay, I'm allowed to import a configuration from a configuration file. Now, there's a configuration file editor that will let you change that value on a GUI but since it connects to an XML file you can bring it up in notepad and change it if you need to.

Carl Franklin: Right.

Richard Campbell: So when you're running these packages, I can just see that they would run on -- since they're running on the server, they might be remotely invoked and so forth. What about the reporting of the results, where does that turn up?

Kent Tegels: You can log at to basically any place you like. You can even write your own log provider if you're using something like Unicenter. My tool of choice is to log it to the SQL Server database that I'm writing at too and the log files can be pretty, yeah, you can get a lot of rows in that but with SQL Server it's pretty easy to filter out the events that you're just looking for and I suppose really if you wanted to you could use the reporting controls in Win Forms or you could write in SSRS report to do that.

Richard Campbell: Yeah because of course naturally it's tied to reporting services in some way.

Kent Tegels: Yeah. The big trick there is you've got to remember to configure logging and you basically have to make sure that every event you're interested in you've checked because it's off by default.

Richard Campbell: Oh, I see.

Carl Franklin: Ah.

Kent Tegels: So yeah, the don't -- because logging has overhead and as the point of the SSIS is to get things done as quickly as possible. They did a lot of logging and so the package has done a

tremendously. So they gave you a lot of control over what is logged but it's your responsibility to make sure that you tell it what you want and what you don't want.

Carl Franklin: This portion of .NET Rocks! is brought to you by our good friends at Red Gate, makers of ANTS Memory Profiler. You know what a Performance Profiler does. You run it and it tells you where the bottlenecks are in your code. You can profile any .NET application including ASP.NET web apps. So if you're a .NET developer of any kind and you want to find out where your code is choking, go to www.shrinkster.com/19op, that's one, nine, the letter O, P as in Paul, and check out Red Gate's ANTS Performance Profiler. You'd be glad you did.

Richard Campbell: I feel like we almost have a missing piece here. I could see a standalone application built by developers that wants to say invoke data imports. Say there's data being brought in through FTP, or some kind of WCF connection, or via the web and as it arrives an application notices it, it's got some kind of watch and it sees it as it's coming and it knows what package to invoke in SSIS to actually import that data and make it available, and you can do this all very cue-based sort of asynchronously. The challenge would be what am I communicating to on the SSIS box?

Kent Tegels: About Service Broker.

Richard Campbell: Right, right. Does this still exist?

Kent Tegels: Service Broker still -- Service Broker has not gotten away and I don't think it ever will because they use it for too many for the internals of SQL Server.

Richard Campbell: But it sure seems like the red-headed stepchild of SQL Server.

Kent Tegels: Well, I think WCF has taken it off the developer's pockets and that's kind of sad but I think Service Broker is one of the very, very cool things that they introduce with the SQL Server 2005 and it just does not get enough attention.

Richard Campbell: Right.

Kent Tegels: But I love to come back on and do a talk about Service Broker because it is so cool.

Richard Campbell: And there is a 2008 version of it.

Kent Tegels: Absolutely.

Richard Campbell: So that's really the thing to call if you wanted to invoke into the SSIS correctly.



Kent Tegels: Well, it's one way to do it. There are lots of ways. I think one of the nice thing in general about the Microsoft platform is you have the ability to code whatever you can code and that gives you tremendous flexibility, but as a lot of people have been on the show said it's just figuring out what you wanted to do and figuring out what's available to do it with because there's so much stuff to pick from.

Richard Campbell: Right.

Kent Tegels: Those of us who have a SQL Server bias, we're going to look towards SQL Server first. Those who don't have SQL Server bias even look at the framework. Nothing wrong necessarily with either ways. There's different ways of looking at the solution stacks we know.

Richard Campbell: And the Service Broker is a cueing stack so immediately I think MTS as well.

Kent Tegels: You could do MTS. You could probably even do WCF.

Richard Campbell: Yeah.

Kent Tegels: You can certainly set up a WCF service that takes a message saying go invoke this package.

Richard Campbell: And it just runs on the SSIS box listening on some kind of port, you push your thing in and off it goes.

Kent Tegels: You bet.

Richard Campbell: That merely begs the question for me of security, but I'm afraid I certainly I can't put my IT hat on here because if you say security on a dev show everybody walks away.

Carl Franklin: Right.

Kent Tegels: Well, I'm not so sure that that's true anymore. There are lots of developers who will still shy away from it but I'm really been blessed here as I'm starting out here at CPU to work with some really brilliant security people and a lot of them are developers, a lot of them have a development background. So I think security has certainly come to the forefront and they think there are a lot of different ways that security can be done with things like WCF. In this particular scenario, it wouldn't be hard to send the message encrypted. You know, basically the encryptor would have to have a key to encrypt the message to send you to invoke for the work.

Richard Campbell: The big thing I'm concern about is what privileges that SSIS service has to have to

execute this package. I mean, obviously it needs to read from a disk, it needs to be able to write into the database and vice versa perhaps. That to me seems like pretty significant privileges for a non-human interface account.

Kent Tegels: I'm running SQL Server 2005 and 2008 under network service so that's a fairly low privilege account to start with, but yeah, I mean you could really tighten it up if you needed to. But the thing I would worry much more about in a service-based scenario would be denial of service attack, just trying to take the service offline by overwhelming it with requests.

Richard Campbell: Yeah and hammering it away, dumping bad files up there and having them invoke like mad.

Kent Tegels: Yeah, or even potentially files that are infected with viruses and scripts.

Carl Franklin: Right.

Kent Tegels: You have to be really is -- it's something you do have to think about.

Richard Campbell: If you're allowing people to remotely execute code on your machines, you get what you deserve.

Kent Tegels: But why do we have WCF at all? Because that's, you know, basically owning the same thing. You're writing a service.

Richard Campbell: It's really a question of execution versus, you know.

Kent Tegels: Uh-hmm.

Richard Campbell: You actually want to do with it.

Kent Tegels: Right.

Richard Campbell: We got very geeky there. Lot of differences between SSIS 2005 and 2008? 2005 was really the first version.

Kent Tegels: Yeah. I think you could say that the difference between DTS and SQL Server 2005 was the difference between a lightning bug and lightning bolt. There was just that much difference between them. 2008 really seems to me to be a lot of refinement work that was done. The addition of C# and scripting, the ability to have references to web services without having to use a proxy generator, the new cache component, the new data profiling component, and dynamic CPU scheduling. Those alone are the price of admission. Are they show stoppers? Are they things that are going to get a lot



of attention in the developer community? No. Should they? I don't know.

Richard Campbell: 2008 R2 just shipped.

Kent Tegels: Yeah.

Richard Campbell: I mean, I hate the naming scheme but that's really another version of SQL Server.

Kent Tegels: As I look at it it's more upgrades. One of the bits that I'm really interested in in 2008 are the Gemini bits.

Carl Franklin: What's that?

Kent Tegels: The Gemini bits are sole service BI. Microsoft has got us into the boat of writing reports as developers and I realized, oh, that's a bad use of developers because developers are expensive. So let's figure out a way to push that down into the information worker. That's what Gemini is all about as I see it. It's really their attempt to get Business Intelligence enhance users with a minimum developer involvement. There is something called Project Madison which to me appears to be the result of Microsoft acquisition of Data Allegro for making very large scale databases. Now, whether that's going to be in 2008 R2 or whether that's going to be in the next generation, next release of SQL Server I'm not sure at all, but Microsoft is actually getting the bits out and letting people see them.

Richard Campbell: Although it's a whole other show I do think that Microsoft, with the whole BI stack that's in SQL Server and as it has been evolving, is bringing BI to the masses. It used to be a quarter million dollar guys in white lab coat proposition and now it's pretty much any desktop machine. You've got a chunk of data, let's analyze it, let's do some data mining.

Kent Tegels: Yeah, I can remember trying to do this stuff back in the SQL Server 2000 with their OLAP services and really struggling with the writing MDX and I'm just amazed at how far they've refine it. Now I hardly even have to write an MDX query so they've done a really good job on making it approachable by the masses, and I think they've done a good job of making it much more approachable by developers.

Richard Campbell: So an SSIS package can be checked in TFS so it can be source controlled because in the end it's all XML which also makes it very portable if I want to run it on different machines with the config file. I'm just trying to think of the things that developers generally worry about around using SSIS.

Kent Tegels: Just to close the gate on that one, it also works with SCN. I've set up Subversion and I use it with that.

Richard Campbell: Yeah, you can't resist Subversion. It's just a little too easy.

Kent Tegels: Yup, absolutely. I agree.

Richard Campbell: Whenever you go in, you go in and modify that XML package. You are doing that through the Management Studio for SQL Server.

Kent Tegels: Whatever editor of choice you want. Because it's just an XML file, you could use notepad, you use TextPad 32, or gedit.

Richard Campbell: So is this actually XML that you can make sense of?

Kent Tegels: I can because I know what the schema is. Somebody does need to publish a really good schema reference for it. For the most part, they've done a good job of choosing intuitive names like Connections. Strings have a Connection String property that literally has the Connection String to it. There are some that are little more esoteric than that but for the most part, yeah, it is living the XML mantra of be self-evidencing as to what your intention is.

Richard Campbell: There's plenty of XML out there that's pretty impenetrable. Much of it have been created by Microsoft.

Kent Tegels: No comment.

Carl Franklin: No comment.

Kent Tegels: I'm not going to put my foot on that one again.

Richard Campbell: Oh man, what was that line? The itchy, scratchy chafing of XML, something like that.

Carl Franklin: Sharpen -- what did he call it?

Richard Campbell: The pointy elbows?

Carl Franklin: Pointy -- I can't remember. Pointy, scratchy.

Kent Tegels: The XML gets such a bad rapt of it. Well, it's okay, yeah, it deserves it.

Richard Campbell: You've talked about that whole Lego interface. You're only going to get that from the Management Studio when you load the XML up through that.



Kent Tegels: True, the Business Intelligence Design Studio which is really just Visual Studio. What it amounts to is in Visual Studio the SQL Server team created a VSIP style add-in to do the design of DTS packages. So it's really done in Visual Studio. It's not done in Management Studio at all.

Richard Campbell: Okay.

Kent Tegels: Right now for legacy purposes, in DTS packages there is a plug image you can get from Management Studio that will let you view, and I think it lets you edit DTS packages although my experience has been you wind up rewriting DTS packages anyway. You don't operate them, you see right them.

Richard Campbell: Right and it just did. They're too finicky.

Kent Tegels: Yup and it just doesn't translate very well. You don't go from a design pattern that was essentially extract load and transform to one that's a different pattern of extract, transform, and load.

Richard Campbell: Right.

Carl Franklin: Is there anything that you want to see in the next version of SSIS?

Kent Tegels: G Special Data Sources would be an absolute boom for me personally because they do so much work and that kind of stuff.

Carl Franklin: Right.

Kent Tegels: That would be a great one.

Carl Franklin: Because we have special data types now, and SSIS doesn't work with them?

Kent Tegels: It doesn't understand them as anything more than a stream of bytes. The real problem for me is that it can't read shape files and it can't read a lot of the data files that we get our geographic information in for defining those polygons. So that's the reason why, as Bob talked about, when he was on your show he talked about a tool called FME from Safe. It's very SSIS-like but it knows how to read those final formats and it's an absolutely great tool.

Richard Campbell: Yeah. I'm thinking about all the data you can get out there today that is GEO special data and you get back to that same old problem, parsing this yourself is a bear.

Kent Tegels: Oh, absolutely right, and pretty much anytime you have to parse data yourself. It's a bear. SQL to SSIS does a good job of letting you write scripts as data sources, write scripts as data destinations, and write scripts as data transforms. All have quite a bit with that, but there's still a lot of processing to be done and still a lot of programming. I mean, in that sense the programmer part in me is very happy because I can go write code and enjoy that experience but I can still do it taking full advantage of a nice framework for data input, data output, and data true put. But I see some different visualizations I'd like to be able to take. For example, here's a chunk of XAML that describes how I visualize data and be able to do some 3D rendering of data with it. I think that would be very impressive. Those are bits and pieces of things I'm looking for. As always, I want more true put, I want faster processing time and all that kind of stuff and Microsoft is usually pretty good about getting that done. To me it's improving the reach of the product for data source, maybe looking at a few more data destinations if they're really needed, it's hard to say, but giving me some more visualizations at runtime would be really nice.

Richard Campbell: All right. Have we missed anything on SSIS?

Kent Tegels: I don't think so. I think we've pretty well covered that list.

Richard Campbell: Those are the things that devs need to know.

Carl Franklin: So you're speaking anywhere? Coming up here?

Kent Tegels: I'll be talking at HDC, the Heartland Developer's Conference in Omaha in October. That's a show that I try to make every year. There's a couple of years where I couldn't make it as I was on assignment with Microsoft, and Don came for me for a while so it was fun to have Don Demsak and be interchangeable with me for a while although the irony of that is pretty interesting. I'm looking at maybe doing a couple of code camp type things in Minnesota and we're trying to get a .NET users group going here in Sioux, South Dakota. I'm looking forward to getting back with Mike Milinkovich and doing some talks with him.

Carl Franklin: Cool. So why is Perl site so cool? You can tell this is a loaded question.

Kent Tegels: Oh boy, is that a loaded question.



Kent Tegels on SQL Server Integration Services September 22, 2009

Carl Franklin: This is like a Perl site day for us. We're recording you and Aaron on the same day.

Kent Tegels: Oh wow, cool. Actually Aaron is one of the reasons that I think Perl site is cool. Many years ago, I think it was in about 2004, of course that's many years in this industry, it hurts admitting that to myself, I went to work for a company that Bob have recruited me into. Bob and I got to work together, but before I had gone to work for that company, I'd been a student of theirs for a long, long time and I had classes with Fritz and with Aaron and with Don Box and all of these big rock stars and I was always just blown away, and I remember sitting in class one time and I'm going wow, if I can ever be 1.0% as good as Fritz Onion I'll be really happy with my career, and then here in 2006 I was back in the same classroom teaching, a revised version of his class going, holy cow, I've made it this far. Well, you know, those people left that company and they went to form Pluralsight and earlier this year when my relationship with that company came to an end, I was looking for any home and the folks at Pluralsight were gracious enough to contact me and say, "You know, we'd really like to have you on board to help us with SQL Server." It was just a no-brainer. Just getting the work with those people, it's just been fabulous.

Carl Franklin: Yeah.

Kent Tegels: When you work with the best, you're challenged to be your best.

Carl Franklin: True. So how did you get into databases and SQL Server?

Kent Tegels: It's kind of a bizarre thing because I was trained as -- when I first went to school, I was doing economics. I always thought, ah, I want to be an economist. Well, I found out two things very quickly. You have to know a lot of math to be an economist.

Carl Franklin: Yeah.

Kent Tegels: And if you know a lot of math and you know a lot of economics, you don't have any job to show. Really you go to work for the Federal Reserve, but at that time there wasn't a lot of stuff to do so I was able to just take that knowledge of data manipulation, and working and storing data inside the databases and manipulating it and just got really into databases from there.

Carl Franklin: That's cool. Hey, we're just about out of time. Are there any last minute things that you want to throw out there?

Kent Tegels: You know, Carl, I'd love to have a show where you get to talk to Sacha Behn

from the UK because he is a really good SSIS person and he's got a lot of funny stories to tell so I think that would be fun if you could do that, and please, please, please bring Don Farmer back on the show.

Carl Franklin: Oh yeah.

Kent Tegels: I really like Donald and he is just great to listen to. So if you can pull that off for me, I'd be a very happy .NET Rocks! listener.

Carl Franklin: Great. Yeah, that sounds within the realm of reason I think. Don't you, Richard?

Richard Campbell: Absolutely. Just recently had contact with Don. Let's see what we can do.

Kent Tegels: That's great.

Carl Franklin: Kent Tegels, thank you very much. It's been enlightening.

Kent Tegels: Thank you gentlemen.

Carl Franklin: And we'll 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.