

Executive Summary

- Site architecture is a combination of understanding the technology that you are working with, understanding the types of issues and problems that come up with it, and merging the troubleshooting and problem-solving skills and techniques, along with Web-based technologies, to deliver a streamlined, easy-to-use experience on TAC Web.
- You can have good results by starting simple and posting some Frequently Asked Questions.
- The more advanced Web sites, such as Cisco's, are providing special tools, diagnostics, and capabilities that assist the customer in troubleshooting, as well as providing online documentation.
- Cisco does not quickly jump to new Web technologies to present technical support information. If you are providing tech support to a large number of people all over the world, you don't want to have to require your customers to have the latest, greatest browser version and plug-in in order to find an answer to a simple question.
- The next challenge Cisco's TAC Web faces is keeping on top of an ever-increasing depth and breadth of information.
- When you look at e-support as a whole, the next advance will be taking the human out of the critical path. If a device knows that there's something going on that's wrong, and can give an indication of what's wrong, that indication will go through the Internet back to the support Web site. At that point, diagnostics and online troubleshooting can take place at a machine-to-machine level, without necessarily needing a human as part of the process.

Site Architecture

Site architecture is all about looking at TAC Web from a very high level, looking at the big picture, and then getting all the components to work together to solve customer problems.

Randy Jewell is a technical manager in TAC Web. Randy's job includes both the high-level thinking about the overall direction of TAC Web, and the nuts-and-bolts implementation of that direction.

**Randy, tell us a little about your background. Where'd you go to school?
Jobs before Cisco? Jobs at Cisco?**

I graduated from Northwestern University with a B.S.E.E. After I graduated from college, I was at Motorola and was a firmware engineer for central office switches for the first cell phone networks that were put up. That was neat, so we were kind of a group in the back of Motorola just cranking out, trying to race to be the first cell phone providers. At Motorola, I really got addicted to the rush of developing new technology.

I did that for four or five years. Then I went to work for an old friend in the EDA (Electronic Design Automation) industry and worked in a field office as a field application engineer. I spent a couple of years working directly with customers, doing a lot of the pre-sale side. I was the benchmark guy, and we go in and try to do all of the technical analysis of what customers were looking for, trying to close the sale. I wasn't an account manager so I didn't have to do all the negotiation stuff but doing the selling on the technical features, doing all the demos, and then doing support also—kind of the frontline support like an applications engineer would do—whenever they get stuck, help them out, get the software working for them, help them with their design. This was really fun in that my job was getting potential customers excited about a new technology.

I tended to focus on ASIC (Application Specific Integrated Circuit) and digital logic design. I did that in the field for about three years with a company called Daisy Systems, then moved to the their corporate headquarters, which is how I got to California. I became a corporate application engineer; I was responsible for a lot of the technical marketing roles of their digital logic simulation products; which features should be in there, how should it be done. We were the ones who created all of the demo databases, demo flows, and demonstrations and trainings.



“I try to keep their (the customers’) networks healthy and try to translate that into a set of services and offerings that deliver what they need in an easy way.”

From there, I went to a startup that was named Synopsis that focused on an emerging field called logic synthesis, which is the idea that you could write programs that, instead of running on computer processors, would actually be the processors and hardware at the back end. My role there was, again, working with customers in understanding their design issues at the ASIC level, and helping define the language and the grammar of the language that we would use that would then generate hardware circuitry. It was actually based on some standard languages which we defined into subsets and language constructs and how would certain language constructs generate certain hardware components.

Again, you can see my passion is in making emerging technologies useful. Listening to customers’ problems, and enabling them to take advantage of the latest techniques.

I did that for seven or eight years, and then came to Cisco. I came here because I saw the Internet as the next emerging technology to dramatically increase productivity. I joined Cisco in the IT group because that was, for the most part, the part of Cisco that was doing Web-based application development, which was my interest. It was the first and only time I’ve ever been in an IT organization. But it was the idea of coming here and again, matching what customers are looking for and trying to deliver Web-based applications that would help them service and support their networks and their devices. So it was kind of neat. It was being able to take all of those past experiences and just turning them into automation, essentially, and trying to automate the troubleshooting and debugging process.

I managed the ACS (Advanced Customer Systems) group of developers, about 20 engineers that were developing the applications. I did that for about 2 years and have now since moved over to my current position, which is site architect. It’s more focused on not necessarily how I build things, but the “whats,” what gets built and why. And so I’m not directly responsible for managing people anymore as much as designing and developing some of the services we’re offering, and the motivations for why we’re doing it.

Could you expand a little bit more on what a site architect is and what a site architect does?

I see my role as delivering solutions to customers’ problems using the most appropriate technologies. I learn what the customers are trying to get done, the types of problems that they encounter, and what they see on a daily or weekly basis on their networks. I try to keep their networks healthy and try to translate that into a set of services and offerings that deliver what they need in an easy way. So it’s a combination of understanding the technology that they’re working with, understanding the types of issues and problems that come up with it, and merging the troubleshooting and problem-solving skills and techniques

along with Web-based technologies to deliver a streamlined, easy-to-use experience on our Web site.

Basically, it's trying to, again, understand what they need and how we can best deliver it in an automated fashion that's obvious, easy, and intuitive. It's understanding the customers and being able to get them the best applications we can, or the best troubleshooting experiences, the best technical support experiences as we can on the Web.

When you look at the Web generally, what are the most advanced Web sites doing today? In both support and e-support? What's the state of Web technology today?

I think you see companies falling into a couple of different categories. The basic level, which is probably the most prevalent, it's kind of like an online library. It's just a collection of information, Frequently Asked Questions (FAQs), maybe some software drivers, that sort of stuff, which people can come and read to solve their problems. There's still a lot of sites, even technology companies, that don't have very sophisticated online support. It's kind of amazing sometimes.

But if you look at the companies who really get it and the ones that are really pushing the envelope, you see a couple of things. One is the more advanced Web sites are providing special tools and diagnostics capabilities that assist the customer in troubleshooting their problems, and that's the space that Cisco fits in. We have more than just documents and knowledge, we provide tools and guides and so forth to help you through the process.

Another common service many companies provide on the Internet is tools to automate common processes, like how do you create and interact with the corporation, even though it's real people that you might be interacting with. Automating the process of opening up trouble tickets, communicating during that troubleshooting session, and resolving it. And then you get some people that are really pushing the envelope where they're actually starting to connect the end devices back to the centralized knowledge-bases through the Internet. And it's not through a Web browser, but a more direct Internet communications.

For instance, many of the PC makers are in that space now where they have some troubleshooting or technical service application that they ship with their PCs. They run diagnostics on the PC and send messages or communicate back to the main Web site to get updates. You have probably seen this in action in Microsoft's applications. For example, when the Web browser crashes, it says, "Do you want us to send all the information to Microsoft?" It doesn't help you troubleshoot, but it helps them identify common recurring patterns and address them.

There's a vendor out there named Motive Communications that provides the core technology many vendors use. It contains an application running on the PC, which can check for common problems with a computer or its applications and advise the user on how to fix the problems. The application keeps itself current by updating itself over the Internet. If a problem is

encountered it cannot correct, then they gather relevant information from the PC and send it to the support provider to complete the diagnostics. It is a really good example of how to use the Internet to provide support to even very basic users.

Today's real state of the art is that the electronic support doesn't have to be browser-based; it's Internet-based. So it's browser-based in most cases, but you really push the envelope and you're starting to get actual devices under monitoring to Web site connectivity.

When you think about companies who are doing that, aside from Motive, any other companies as examples of having really good sites?

Yes, actually I should probably clarify my last statement. Motive in-and-of itself doesn't do it. They provide the software that a lot of other companies use. For instance, HP, Compaq, Dell, Gateway, all of them are doing things like this. And it's pretty cool. Personally, I just bought a new HP computer recently and I've just been blown away that I can buy a computer at a retail outlet and whenever I have problems, I click on a button. Boom! It's firing off a chat session for no cost to me to HP. First, it tries to do all of the automated diagnostics, which sometimes figures out what's wrong and it doesn't even have to go to HP, but when it does, it packages everything up and—Boom!—I start getting a messaging session going with them and you know, that's just what I got out of the box.

When you look around and try to find the latest Web technology, how do you do that? What do you do for research?

There are a couple of things. One is staying up with trade magazines. Actually, I should say trade Web sites now. I don't read the magazines much anymore. They just kind of pile up on my desk. But it's a combination of staying up on some Web sites that stay up on the technology. One of the other things is, being at Cisco, we're in a little bit of a unique position in that a lot of technology comes to us and a lot of small companies who were really trying to make it in the Internet space would love to come work with Cisco.

So within the IT organization, when I was there, I mean we get calls every day or every week to say, "Hey, I'm a new company that's got this hot new idea. We'd like to come present it to you and see if you want to work with us." And so we get exposed to a fair number of companies and companies' technologies before they're prime time because they've got an idea, they've got a core kernel going, and they'd love to partner with us to finish fleshing it out. So that works out pretty well.

Otherwise, it is just keeping a finger on the pulse—you know, the general hoopla in the media, What are the big technologies? What is and isn't working, and so forth?

Personally, I tend to focus a little less on a technology acronym and spec as much as I do the application of it. So just because SOAP (Simple Object Access Protocol) comes out and

it's a real hot protocol, it's interesting to know what it is, but it really becomes interesting to me when it matures to where I can actually start doing something with it.

Is the Cisco TAC Web site cutting-edge in terms of technology?

I think it is in many ways. It definitely uses a lot of the state of art in terms of Web development techniques and some of the leading edge diagnostics and services that we offer. At the same time, the leading edge is coming closer to the mainstream on a daily basis. Within the last two years or so, the general industry, not just the networking industry, but the general support industry has realized, "Wow, we can deliver a lot of support on the Web." And so where maybe three, four years ago, we were like several light years ahead of the rest of the industry, now a lot of it's caught up because a lot of the initial steps are pretty straightforward to do.

I think there are, of course, always things we can do to improve it. There are things that are both technologically within reach today as well as out of reach and things we dream about.

How do you decide to upgrade to new Web services and functions? And when do you stick to the established stuff?

We don't jump to the latest technology just because it is hot. The reason is because we are providing tech support. I don't want to have to require the latest, greatest browser version and plug-in in order to find an answer to a simple question. So in many ways, from at least the client level or the browser level, we don't necessarily want to be the hottest, latest, greatest, XML-rendered content if nobody can look at it. So we tend to be a little slower maybe there. But for a very specific reason.

We started out pretty much as a Perl-based shop, doing almost everything in Perl. Then Java became the big, hot language. But that didn't mean we switched from Perl to Java. While there are many benefits to the language, it has to be considered relative to the experience of the staff and the other resources that can be shared. When creating applications on the server, Perl and Java are two tools we can use. We pick the one that is best suited for the project at hand. To this day, we are a mixed shop, with applications being developed in both languages.

But when you see new technologies, like some of the push technologies, whether it's like a Tibco, or a Backweb, and when and where a new technology like that makes sense, we deploy it in a controlled fashion, where there's a particular application or service that says, "Hey, this would be a really awesome way of delivering it." We will implement it, see how well it performs, and how well the customers can take advantage of it.

One of the things that we always keep in the back of our mind is that, because of our success, the site is an extremely high-volume site.

So given that we have a huge installed base and a huge expectation level by our customer base, that tends to temper us from deploying a new technology until it has proven its stability. If you're a start up and you're getting your Web site up and running, yeah, you might decide to go ahead and build it in the latest, greatest, hottest technologies because you're customers say if it's up that's fine, but if it's down, well, that's okay, because it wasn't there before. But Cisco is doing over 80 percent of its revenue through its Web site, so you don't want to mess around with something that's going to interrupt that flow.

How do you decide whether to build something new in-house or purchase it from a vendor?

Well, I'll give you two answers. Historically, we've always built in-house, and the reason is, being an early adopter, there just wasn't anything there. The philosophy today, though, has changed. The market for Web-based applications and middleware has matured over the last couple years and many vendors and tool developers have jumped into the space to offer different components. Buying makes a lot more sense, many times, than building does. And so we have been going through a change in terms of evaluating build versus buy.

I think the first step you take before you can make that build-versus-buy decision is make sure you focus on what is it you're trying to deliver. What is the customer need? What service are you trying to provide? And what does it need to do? That's a step that's easy to gloss over because we don't carefully articulate or carefully define what it is we want to do, which can lead to the downside of jumping to a hot technology just because it was there and it was a good demo.

But the key is make sure you define well what is it you want to do and then work with the IT group to go out and say, "Okay, who sells something like this and how much and what's the reliability and what would it cost to build internally?" One of the things working against us a little bit is the fact that we have a huge Web environment today and Cisco is so interconnected and so Webified everywhere that one of the prime considerations ends up being integration to Cisco. A standalone application that works in and of itself, all by itself, very well is in many cases a step backwards for us because we want it to be integrated with the various different work flows and knowledge flows within the company.

So a lot of times, one of the main decisions, after we figure out what it is we want to do, is figuring out if it fits within the Cisco infrastructure today. And I'm not just talking does it run on our Web server with our Web server technologies, but does all the information and knowledge that is needed get to it? Does it flow in an automated fashion or through some sort of interface, and can we integrate all the reporting back to our reporting schemes, and

can we do all of those things? And since so much of that infrastructure is home-built, that ends up being a challenge for off-the-shelf software many times.

Where is Cisco's whole Web site going? And how does TAC Web fit in with that as a result?

What you're going to see from a Web site point of view is a much more polished or integrated, streamlined feeling. One of the criticisms of the Cisco Web site, which I think is very valid, is that it is not organized well, and that's not a technology issue, that's just a process and information architecture issue. So I think what you'll see in the next steps is trying to get a handle around the organization of information, making the site just behave the same no matter where you are on the site. So you should always feel like you're on a Cisco site.

I see that the next step for Cisco is really information architecture and process. It's getting everybody using the same terminology—to call the same things the same thing all the time, every time, in all the same places. TAC Web fits in there in a very major way. Looking at the various different analyses we've done, it looks like the people coming to our TAC Web site to do technical support, basically to look for our information, is roughly a third of all the traffic that comes through the whole Cisco site. So we need to be a very tight part of that, the common vocabularies and common organizations of things. The things we come up with need to be ones that make sense to our customers as well as to a customer maybe considering a purchase of a product from Cisco, or for that matter, somebody considering investing in Cisco. All of those types of people need to be able to come and find what they're looking for.

So we're very active in working with getting those kinds of conventions and standards and process all adopted across the company. You know, recently, you've seen one of the external effects of that. We reworked all of the applications that we have on the site, so that they all have a very common look and feel. Before they all kind of did their own thing their own way and now, if you go and look at almost every one of our applications, it all looks the same, behaves the same, has similar buttons in the same places. And that sort of consistency is really one of the keys we need for the next level to keep things predictable from a customer's point of view. While every Web site on the Internet seems to do things differently, we need to ensure our site is consistent throughout. This way, customers can learn how to use one portion of the site, and then use that knowledge when interacting with other portions of the site.

Where do you think Web technology is heading in the future? And how is online support going to change because of that?

As I hinted at earlier, I think when we were talking about some of the leading edge stuff, I think we'll take the Web browser out of the picture in many interactions. What we really want to get to, if you think about it, is a scenario of either troubleshooting or taking

corrective action when a device has indicated that something is amiss. Today, we then have a person, the network operator, who sits inside and goes, “Okay, there’s something wrong with this thing. Let me bring up my Web browser and try to troubleshoot it.”

I really see that the next step is going to be taking that person out of the way, or at least not be in the critical path. If the device knows that there’s something going on that’s wrong and they’re able to give an indication of a problem, that indication will start hooking through the Internet back to a service that can perform diagnostics and online troubleshooting. This can take place at a machine to machine level, and not necessarily needing a person in between.

And then you’ll be able to see corrective action that can be taken automatically or by a network administrator. Of course, nobody really wants their network changing without them knowing, so you include the network management teams and the administrators involved in the corrective process, not necessarily during the trouble-shooting process. The machine to machine should be able to do the diagnostics and troubleshooting and then wait for the administrator to actually take the corrective action to fix things. Technically, as I see it, we could even do the corrective action directly with the device but that will be based policy-wise on whether or not the administrator is going to want to do that or not want to do that.

How do you determine what customers do on the site? How does that change the design of the site?

We determine what customers are doing in a couple ways. First, we talk to them. We run advisory boards around the world where we’ll ask them to come and tell us how things are going. We may go into the specifics set of objectives that we want to learn about or sometimes just go in and listen to what they have to say. We will test applications that way as well. Maybe we’ll have some new applications, run them by target audiences and see what they think. We also will work on them almost one on one. We’ve been doing direct customer visits where there’ll be one or two people from Cisco and we’ll go sit out in the NOC (Network Operations Center) of a particular customer and watch them for a day or two and see well, what is their process internally? How do they figure out when they need support help or how are they offering support within their own company.

When they’re delivering their support internally, when do they need to work with Cisco as a vendor to help them? Is it through an online experience? Is it through calling in the TAC? So we’ve actually been trying to just go out and watch them, understand their behavior, understand how they’re doing things internally so that we can see how they interact with Cisco.

Secondly, we do a lot of statistical analysis, too. We have a fair amount of statistics in terms of Web usage. Who comes on what days? How often? How long do they stay on the site? We can look at the click stream information, which is every URL they click on in which

order. We've tried to see if we can identify patterns. Who's doing what when and what's popular, and what's not.

We also then try to work with our TAC organization to understand what types of issues customers phone in. Are people calling the TAC for things that are the same, or different than what they can get from the Web site itself. So we kind of look at all those different areas.

How well do you think Cisco's TAC Web is organized? Can people find what they need?

That's a loaded question, right? We've probably made it more difficult than it needs to be. And that's why, earlier, I was talking about when we talk about where do we see the entire site going, a lot of what I talked about was information architecture. Internetworking space in general has been a very hot emerging field in terms of new standards, new specs, new capabilities, new devices popping up all the time. Keeping up with that has been a challenge, and I think that in our dash to keep up with technologies and product offering in the networking space, we haven't necessarily always done the best job of information architecture and organizing things as well as we could.

It's definitely not organized as well as it could be, but we're recognizing that. It's like I said. I think the next step that we're going to be focusing on is trying to make sure that our customers understand how to get to all the information and all the resources we provide.

When the average customer comes to TAC Web, do they usually use search, or do they usually use navigation to find what they want?

People usually fall into one of those two camps. Some people like search, some people like navigation. On our site, more people search than navigate. It's about two-thirds of the people search as opposed to navigate. But when you start watching them and talking to them and understanding what they're doing, you'll realize that, many times, a lot of those people are searching because they think they cannot navigate. In fact, they would prefer to browse to it because they believe it was quicker, except that given that our site is somewhat difficult or challenging to navigate, they tend to fall back on search. So I know there's some of the usability sorts of gurus out there and some of the thinking and some of the UI design or site design, suggest that if search is used heavily, it's not because that's the preferred mechanism, but it's because your navigation is failing. And I really believe that. I think that if we watch people and how they would like to get to the information, almost everybody you talk to, when they say, what do you want? It's like, well, I just want to be able to click, click, click and get what I need. And that's what they want.

That's not what they're doing, but that's our problem, and it's a solvable problem, too. That's pretty much what we're focusing on. We're cleaning up the navigation so that customers can be successful at getting to the information they want through a series of clicks.

When you introduce a new feature, is there a price to be paid for customer satisfaction before they get used to it? For example, like if the site is reorganized, people might have trouble navigating to a certain place, even though the site is greatly improved. They get kind of used to the old way.

When we do changes, customers almost always don't like it, initially. I mean, nobody really likes change. Once you finally figure out how to get to what it is you're looking for, if you come back and everything's totally different; you're faced with relearning what you just learned, and a lot of that effort you went through is just thrown away. Right? So, yeah, customers definitely don't like us to change it frequently.

Whenever we do make a change, we always get a fair amount of e-mail saying, "Why'd you change it? What are you doing? You change it too much." Whenever I see one of these, I send them back an e-mail asking, "Okay, we're changing too much. How often would you think is reasonable?" And it's actually kind of interesting because the answer usually comes back somewhere around the line of two to three times a year, tops. And about half the people who respond go, "You know, I guess that's about how often you do it."

So it's one of those things that they recognize and realize that it's going to change and it's going to evolve and it is kind of painful but at the same time they realize that it's somewhat inevitable.

Now, that's not to say we should just stop there. I think there are a lot of things we can do to make that change a lot more comfortable or a lot easier for them. If we did get a reasonable information architecture we can minimize the number of changes that have to come. Today we have a lot of change because we just don't have things organized well, and we don't really have a good set of processes to make sure that they stay organized well.

There are probably some areas for improvement for us in terms of helping teach them about the changes, or getting them prepared for change ahead of time. So in some of the changes that we've got going on you're going to see them up in kind of shadow mode for a while, and we are going to tell customers what's coming, ask them to play with it for a while, and let us know what they think about it.

Additionally, there are some things we can do better in terms of training materials. We're going to start placing more "Here's how to use it," or "Where did it go" messaging on the Web site so that they can understand how to perform the same tasks they used to perform in the new ways we offer them. Or at least make the learning curve shorter by teaching them how to use the new things. Yes, change is inevitable but I think we do need to make it a little less drastic, a little bit more predictable as to what the changes will be.

What effect do customer bandwidth issues have on site architecture?

It's definitely important. One thing where I think we're a little different is that our customer base is probably not the norm, and so we don't watch it quite as closely as a lot of consumer Web sites do. A lot of the sites, like a Yahoo!, or C/net, or AOL, where they're getting a lot of low bandwidth consumer traffic, they have to have very small, fast pages because their customers are predominantly dial-up customers.

Our customers tend to be large corporations or service providers and, for the most part, have reasonably high-speed connections. So it's a little less constraining to us than perhaps a traditional, or more mainstream Web site. Which isn't to say it isn't important. But I would say that we do need to watch it extremely closely.

How do you use the metrics in site architecture? What do you look for when you get those metrics reports?

I use metrics to understand the customers' behaviors on our site. I tend to focus on usage trends and patterns as opposed to a specific user's actions. One of the most direct metrics is feedback from our customers. We provide places on almost every document for the customers to rate the document. These scores are used to identify documents that are working, and documents that are not working.

Also, we look through the Web server metrics.

We can see hotspots in terms of some of the contents, like which areas customers are coming to the site to learn about, or research about. Those are the metrics I look for. I don't necessarily look at absolute numbers in the trends. It's hard to really make some sense of the number of times a document was hit, for example. What does that mean on its own?

I take the number with a grain of salt in that it's a point in time, and just watch what happens to that point. I really don't care what the number is. So you look for trends, like amount of usage by domain, and then projecting meaning, such as maybe the increased usage in that domain maybe represents a shift in the technologies our customers are using, or perhaps it was some new technology that people are finally starting to learn and get used to.

Or, if it's declining, maybe it's an area that you don't necessarily need to pay as much attention to anymore because the customers just aren't using it. Again, as the technology space continues to speed ahead, there are a lot of old technologies that just don't get used much anymore. So we can use that to guide us in terms of what we offer. If there is an old technology that's still out there, but just not used much, well, we won't invest in creating a lot more information and a lot more knowledge on it. If you see something emerging coming up, we'll jump on it and make sure that we can cover that topic area and deliver the solutions we're looking for.

I think we can also see it if you look at some of the patterns in terms of how they like to interact, too. There's different types of content, but there's also different ways in which

customers like to interact. We can look at the trend there to see if we are seeing more people searching more and more as a trend, or, as a percentage of navigation, is it less?

Who uses TAC Web? When you look at the people who use it, what's their formal education, technical training, knowledge of English, number of years working with technology? Do you look at the background of the people who come to TAC Web?

We conduct surveys of our customers on a regular basis, so we have a good idea of who is coming to our site. Historically, the technical competence level has been very high. The people who are doing network deployment four or five years ago knew what they were doing, or at least were very technical, had very high technical aptitude to figure out network problems and learn and experiment on it themselves.

But we're actually seeing over time that the competency level is going down. As networking as a concept and as a technology is becoming more and more mainstream, we are finding a lot more people who expect to buy a box at a retailer, plug it in and it should work, and that's all they should ever have to do with it. And they're expecting that they could buy ten of them at the retailer, stick them in their company, and the network should all work well, not recognizing that there's additional complexity as the networks get large, and a lot of design considerations should go on.

While we're seeing people who are less technically savvy and certainly less trained in the specifics of networking technology, we've still got the people running the largest and most complicated networks in the world and those users are still using the site to learn things. So, one of the newer challenges for us is ensuring we provide information for both of those types or users.

When you look at the reasons that people come to TAC Web, is it mostly for things like, It's broken: I need to fix it, or do they come for background information? What's their motivation for logging onto TAC Web?

This is shifting a little bit. Being that we are aligned with tech support, almost everybody associates break-fix with tech support, and we've been focused on that. We focus a lot on troubleshooting or corrective action. If it's broken, do this. If something's not working, here's how to tell what's broken and what to do about it. So that has traditionally been our strong suit.

But, many users are not coming to the Web site to look for information in a break-fix sort of situation.

What we're finding is that a lot of people are using TAC Web as research, so if they know that next quarter they have to deliver voice on their network, or they need to do some quality of service implementation, they're using it to train themselves or get educated before

deployment of new technologies and services and products. That's one of the things that we see them doing a lot of.

I don't want to say that they don't use it for break-fix, but it's typically not the really critical problems. If they've got their boss looking at them and saying, "When's it going to work?" they're probably not surfing the site maybe as much as we might have thought. You know, if there's some weirdness that's just kind of an annoyance but doesn't get in the way of the network working, then they're trying to figure it out on their own. I think most people still do like to try to figure it out and troubleshoot it on their own time. I don't know if that's just the engineering way, but a lot of folks still don't want to call and ask somebody, "Gee, how do I do it?" It's kind of like having to stop and ask directions when you are trying to drive somewhere. You don't really want to do it if you don't have to. You'd rather look at the map and figure it out yourself.

A lot of people do have a willingness to try to do the research themselves. If you look at some of the Web activity that goes on, there are users who will sit on our Web site for essentially hundreds of clicks a day. And if you think about how many Web pages that is and how much reading that is and how much researching that is or how much digging around that is, that's a lot of time. That's incredible.

I think you really do see the willingness for them to try to find what they're looking for. And like I said, I think it's shifting more and more towards education, background, preparation sort of work.

Can you think of anything else that you'd like to talk about, that you'd like to tell our readers?

We've learned a few things in the past few years. If you take the attitude that we know the customers are going to run into these kinds of problems, so let's just document the problems and put solutions on a Web site, I think that will get you only so far. One of the things we're learning more and more is you've really got to understand what the customer's whole business cycle is, and what types of things they need to accomplish. What are there challenges with our products and our offerings?

For the most part, Cisco products don't break, that's not really what goes wrong. What goes wrong is they've set it up wrong, so I think if you really start understanding the types of problems customers run into you can do a lot of preventative sort of work. If we can get them to do it right the first time that will make them significantly more successful down the road because they will deploy our product more successfully.

A lot of the really, really easy assistance on setting up, getting things running, verifying that it's working correctly are some of the things we need to focus on, as opposed to a traditional approach of just telling them how to fix it once it breaks.

The other thing I think we've maybe learned is don't overlook the whole usability side of things, the, "If we provide them good information, that's okay. They'll figure out how to get it," sort of an attitude. I think there really is a lot to be said about making sure you build things that are intuitive to your customer base. And that doesn't necessarily mean using the lingo you're comfortable with inside the company, but making sure that you can make the user experience as simple and as easy and as straightforward as possible.

What do you like best about the TAC Web site?

What I like isn't necessarily any particular tool or application or organization or anything. One of the things I think is really important is that Cisco, as a company, is very open about their products and about both the strengths and weaknesses of the products. And I think the fact that Cisco displays all of their known bugs on their Web site is huge. The fact that field notices or technical alerts go out about deficiencies in the product is a really important thing.

I think, in many ways, one of my favorite things about the TAC Web site is the willingness of the organization to put that type of information in the hands of our customers to help them be successful. It is the ability to help people leverage an emerging technology, and become proficient and comfortable with it.