

interview

You'll probably recognize Wil Wheaton as Wesley Crusher, the wiz-kid teen officer from *Star Trek: The Next Generation*. True geeks, however, know Wheaton as more than just a sci-fi character; they know of Wheaton's geeky love affair with technology, his advocacy of open source and free speech on the Internet, and his predilection for retro videogames. Wheaton speaks with us today about his life up to this point, his ultra-popular blog, WilWheaton.net (WWdN), and his new book, *Dancing Barefoot* (published by O'Reilly).

WIL WHEATON



ms tech

08. Can J2EE and .NET Provide Cross-Platform Interoperability? by Dwight Peltzer. This month, Dwight walks us through a brief overview of J2EE and .NET, their architecture, functionality, and available third-party connectivity software.

10. Has Microsoft Finally Gotten the Message? by Terri Wells. MS is taking security more seriously. But given their track record, is that a source of comfort or cause for concern?

OPEN source

11. Rescue Me! by Peter Lavin. Ever been caught without that rescue disk, thinking that everything was lost? Well then, it's tomsrtbt to the rescue! Works with Linux and Windows, too!

13. SCO v. IBM: The Story Until Now by Terri Wells. In case you're confused, here's a brief overview of what's been going on in the SCO v. Open Source ordeal.

hardware

14. x86-64: The Golden Handcuffs by D'Arcy Lemay. What Intel's announcement of the 64-bit Xeon and AMD's move to 64-bit means to developers--other than the availability of terabytes of RAM, that is.

17. Robot Racers: The Agony of Defeat by Terri Wells. NASCAR it wasn't. But for those interested in what autonomous vehicles can do, this desert race was an eye opening event.

SITE promotion

20. On the Right Side of the Law by James Welch. Trying to stay on the right side of the law in Web promotion? Temptation, it seems, takes the form of a number one search engine ranking.

21. Utah Bans Spyware: Feds to Follow by Terri Wells. It's not often that Utah's legislation sets political precedent. But with its latest move, will we start to see the end of Spyware?

WEB services

23. The Web Production Process by Cliff Ritter. Create and implement an effective, efficient, and creatively unique product.

EXTRA feature +

25. Starting a Business, Best Practices to Follow by Romeo Marquez. Pointers on how to successfully build stronger relationships with your clients.

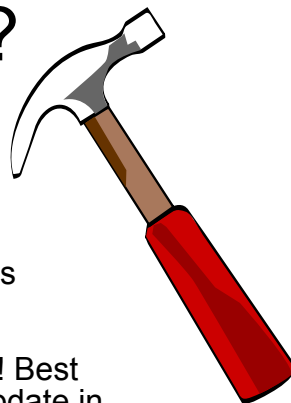
potpourri

- 02. Developer Shed Featured Features
- 15. Want to Be Famous?
- 26. Contest
- 27. grep this *

- 28. Advertiser Index
- 28. Developer Resources
- 29. shutdown -h now

Developer Shed featured features

Looking for powerful tools? Get **DevText**



The Developer Shed Network is now offering you the chance to enhance your site by using Developer Shed's DevText feature, as seen on the right hand side of all Developer Shed pages. DevText allows you to feature Developer Shed's content on your website FOR FREE! Best of all, DevText updates automatically, so content will update in real time!

Why Put DevText on Your Site?

Simply put, the Developer Shed Network offers some of the best technology related education and information on the 'net. Now, you have a chance to make Developer Shed's content YOUR site's content, and offer your visitors industry leading information. Find out more at <http://www.developershed.com/devtext.php>.

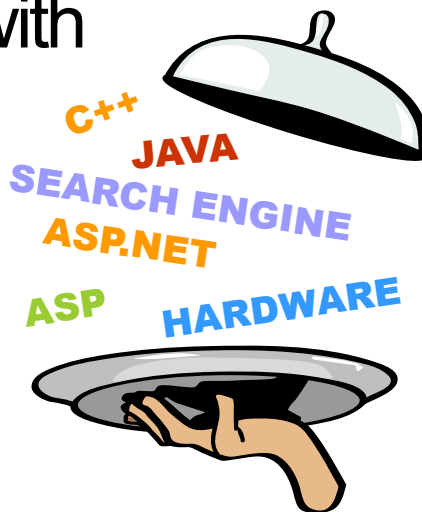


Technology is served with **DevAlert**

DevAlert is the quickest way for you to stay current with information about your favorite technologies. Best of all, it comes to you! Need to stay up-to-date with all the latest info on XML, Apache, and computer hardware? DevAlert will keep the latest information coming to your mailbox, so you get it as soon as it comes out. Get only the information you want, when you want it, delivered to you.

To subscribe, simply go to <http://www.developershed.com/devalert.php>, and follow the on-screen instructions. Within minutes, you'll be ready to receive all your favorite content from the Developer Shed network, as it comes out.

Sign up for DevAlert today!



THE DEVELOPERSHED MONTHLY MAGAZINE

<http://www.developershed.com>

PUBLISHER	Jonathan Caputo
CHIEF EDITOR	Norbert Cartagena
ASSOCIATE EDITORS	Ted R. Long Frank Manno
ART DIRECTOR	Aissa Ovalles
CONTRIBUTORS	Dwight Peltzer Katharine Miller Peter Lavin D'Arcy Lemay Terri Wells Cliff Ritter James Welch Romeo Marquez
SALES	Shari Caputo Michelle Wedderburn

The DeveloperShed Monthly Magazine is published electronically twelve times a year by Developer Shed, Inc.

12555 Orange Dr., Suite #109
Davie, FL 33330
Telephone # (954) 862-1412
Fax # (954) 862-1413

For advertising and marketing information, contact Shari Caputo: shari@developershed.com

For topic and article suggestions, or if you would like to write for us, contact Norbert Cartagena: norbert@developershed.com

Subscriptions are available for free at: <http://www.developershed.com>



INTERVIEW

WIL WHEATON

You'll probably recognize Wil Wheaton as Wesley Crusher, the wiz-kid teen officer from *Star Trek: The Next Generation*. True geeks, however, know Wheaton as more than just a sci-fi character; they know of Wheaton's geeky love affair with technology, his advocacy of open source and activism for free speech on the Internet, and his fancy for retro videogames. Wheaton speaks with us today about his life up to this point, his ultra-popular blog, WilWheaton.net (WWdN), and his new book, *Dancing Barefoot* (published by O'Reilly).



Plug In: Ok, To start from the beginning, give me an overview of how you got started in acting; was it something that your parents pushed on you or was it something that you had decided that you wanted to do on your own?

Wil Wheaton: I come from a show business family. My great grandfather was a character actor and a contract player for Republic Pictures in the 20's and through the 50's. I never knew him, but he influenced my mother because she wanted to be an actor when she was young. In a very odd twist of fate, my great grandfather, at the second half of his life, was a writer. He ghost wrote for one of Harry Houdini's autobiographies and a bunch of other things. It's sort of odd to me that I'm sort of going in the same path. I'm also going to be writing autobiographies for dead magicians!

I guess it was just sort of in the blood because I grew up around that. I grew up around show business; my Mom did a bunch of commercials when I was a kid so I was very used to going to auditions with her. We were really poor when I was a kid so my parents couldn't afford a sitter. I just went with her on auditions and that's what I was around.

When I was 7 years old or so, I showed a little bit of an aptitude toward acting and towards commercials. My parents were real supportive; they never really pushed it on me the way that we would expect for the image of stage parents--you know, where the kid is sitting in the car crying because he doesn't want to go into the audition and the parents are like "You're going to go in there, and you're going, and daddy's going to buy a new suit, you know."

PI: As far as the acting was concerned, what was the most challenging obstacle in being a successful actor, especially if you wanted to be a kid and not go in for the interviews?

WW: Well the hardest thing was when a role would call for me to be really emotional, especially if I had to cry on command. I'd gone to an acting teacher. I was like 7 (it would have been '79. Probably right around 1979, '80, '81--somewhere around there) and he was one of these guys who was, like, wearing a unitard, and wore those 70's sunglasses that everyone wore indoors that were real dark at the top and then got lighter as they went down to the bottom, and they covered up about a third of his face. And he had a big perm and a big... kind of like a Lemmy mustache (you know, from *Motorhead*), and the guy really scared me and really intimidated me. I had been put into this guy's acting workshop at the request of the agent that I was represented by at the time because she thought I had some talent, but I didn't have a lot of skill.

I asked the guy, "Ok. So what happens if they want me to cry for something?" (because I knew that was going to happen sooner or later), and he said, "Oh well, if you have to cry for a movie then they would just put some onion juice or lemon juice in a spray bottle and spray

often called for that sort of thing and my trauma about that and my fear of that almost cost me the role in *Stand by Me*.

They wanted me to cry in the audition and I was like, "Oh, I don't like to cry for movies." Luckily, Rob Reiner was like, "Yeah, maybe you should rethink that." (Laughs) So they let me come back.

PI: So, how'd you get the crying thing done?

WW: When I was younger I cried because I was scared--I was afraid that the lemon juice mister was going to come and spray me. (PI: Laughs). As I've gotten older, and I've worked on other projects, I approach every character from a real truthful, real honest, very real perspective, and I make these characters 4 dimensional, so that they have histories, they have futures, and there's a lot to them. They're not just words on a page, and if the script is well written, and I've done my homework to develop the character well, and this character is experiencing some kind of loss that you know that causes him to be emotional, it just happens naturally.

*"If I could do it differently...
I would just appreciate it more."*

it into your eyes," which is bullshit; they don't do that. But it terrified me, and now I had physical fear, in addition to the psychological fear of being called a sissy, because everybody knows that when you're 9 and 10 years old as a little boy, you don't cry (PI: Right...), 'cause, you know, people make fun of you. And that was the absolute hardest thing for me to do when I was younger, because I was always playing these emotional, and vulnerable, and kind of transparent characters. The roles that I was offered

PI: Any funny practical jokes that you can share with us from the *Next Generation*?

WW: No. (PI: Laughs.) I don't know of any. There may have been practical jokes. There's a thing about practical jokes: it can be a lot of fun and it can break up a lot of tension on the set if that sort of thing exists. Fortunately for us on *The Next Generation* there was rarely tension on the set. Certainly not among the actors. Sometimes we would

“I would probably be really, really rich...I also think that I would be a huge raging asshole...of like Shatnerian proportions.”

get grief from the production office if we were going too slow. I remember once we got a huge lecture from the producers because we were having too much fun and it was slowing down the production because we were goofing off so much. And in television production, it's all about time. It's all about how many pages you can get done in a day. We generally have seven days to shoot 43 minutes of a television [program]. That may seem like it's not very much, but each page in a script is about a minute long. We would be very lucky if we could get seven pages done in a day. If you were doing 9 pages in a day--that was a very big deal. That was a very intense grueling day, and we just didn't have time to goof off.

PI: Except when Patrick Stewart fell through the set and broke it, right?

WW: Yea that cost us thousands of dollars and a lot of time. But that wasn't a practical joke--that was just a fuck up.

PI: If Wesley Crusher were a D&D (Dungeons & Dragons) character, what would his THACO be?

WW: You know, I don't play second edition D&D anymore. I play D20 these days, but I suppose his... I'm going to have to get out my players handbook right here (flipping through D&D player's handbook)...hold on...character sheet... I'm guessing that his armor class would probably be 25 or something like that, and his initiative modifier would probably be pretty high. He doesn't have a high dexterity score, but he's probably got an 18 in intelligence and probably in wisdom, too. God, it's been so long. It's a great question, but it's been so long since I've played second edition--I haven't played second edition since the early 80's.

(After a pause) I have to have a better answer to that. Goddamit!

[Note: *We were seriously impressed that not only did he have the 2nd Edition Advanced Dungeons & Dragons Player's Handbook, but that he had it within arm's reach!]*

PI: In your new book, *Dancing Barefoot*, you said that *Star Trek* was wonderful, and you had forgotten. And you wasted the last 10 years trying to escape something that you loved, and you missed it, and you wanted to get those 10 years back. You wanted a second chance. I'm curious about what you would do differently if you could redo it, and why do you feel that those 10 years were wasted for you?

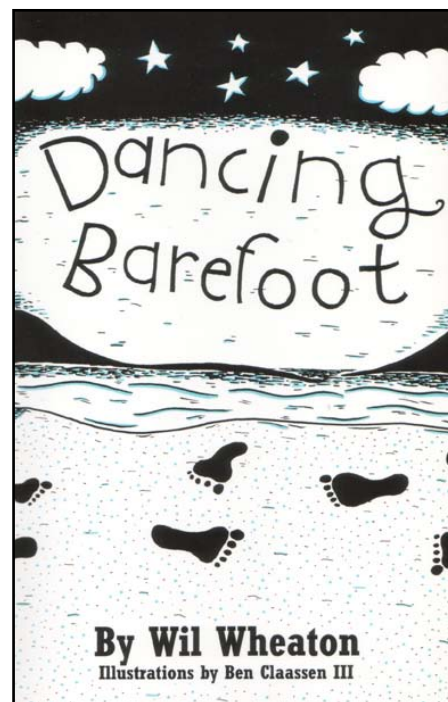
WW: If I could do it differently, I would appreciate it. I would love every moment that I was there, instead of resenting days where I had to sit on the bridge and say nothing. I would appreciate that I'm making a very, very good living doing something that a lot of people would sacrifice their children to do. And I would appreciate the opportunities that I have to be part of something that is going to be bigger than me for the rest of my life. I would just appreciate it more.

Over those 10 years...you know, I think I talk about this an awful lot in *Just A Geek*. I don't think a lot of this made it into *Dancing Barefoot*, but when I was about 14, I went on a *Star Trek* cruise with my parents. It was the first one I'd ever done. The ship was completely filled with *Star Trek* fans; the entire ship was sold out with *Star Trek* fans. I've done a couple since then that were organized by different people and the ship was like 5% *Star Trek* fans, so I can get a little break, you know?

But at this particular cruise it was really like I was at this *Star Trek* convention that never ended, and I couldn't get away. And I remember so clearly walking to the gang plank of the ship. It was the first time I had ever seen all of the original series actors. There was Michelle and George and Walter. But at that time

they were, “Oh my God! There's Uhura! There's Chekov! There's Scottie! There's Mr. Sulu!” And with the notable exception of George, they were all really unhappy.

I mean, I could just see it. They were really unhappy...It was like nine in the morning. Jimmy was drunk, and it was like (in ghostly voice) “Wil Wheaton... This is the ghost of your career yet to come!” And it terrified me. I just thought, “No fucking way that's going to be me!” And I remember I said it. I turned to my brother and I told my 10 year old brother “I'm not going to be 30 years old and talking about what I did when I was a teenager!”



Now, if you've done the math, you'll realize that around the time I was turning 30 years old, all I could do was talk about what I had done when I was a teenager--because I hadn't really done anything worthwhile since then. That really was what killed me, that I had wanted to do all these other things with my life, that I had all this promise as an actor, and I never really fulfilled it.

I took time off from the entertainment industry--naively assuming that I could just pick up where I left off. And that was, of course, wrong. Hollywood is very fickle and runs very hot and cold. I mean, just last year I took 6 months off to write my book, and my agents just



1976



1982



1987



1994



2004

dropped me. I mean they were just like, "Bye!" And that that is really the way things work in the entertainment industry.

PI: Do you think that things would be different for you now if you had stayed the whole time through *Star Trek: The Next Generation*?

WW: Things would be different in a lot of ways. I think that I would probably be really, really rich, and I wouldn't be living month to month like I am now. But that's it. I don't think that I necessarily would have had the desire to breakout of that role, and to go beyond *Star Trek* like I ended up doing. I think that I would have been real complacent, just sitting there, saying my lines and cashing the check.

I also think that I would be a huge raging asshole. I mean really like of--of like epic, *Shatnerian* proportions.

I really think that if I hadn't left when I did... if I hadn't really fallen from grace... if I hadn't really struggled for a long time to determine what was important to me, and discovered a real life instead of a Hollywood life, I would either be dead, or I would be such an insufferable prick nobody would want to talk to me.

PI: So do you regret it or don't you?

WW: No I don't regret it at all, because I'm really happy with who I am.

I would love to be more successful, but that's coming, you know. I'm only 31. I've got a lot of living left to do, and the last few years have been fantastic. Writing this book was something I never thought I would do. I never thought that it would be successful; I never thought people would care about what I had to say. Although I've written stories my entire life, and kept little journals and all that, I never really thought that anything I wrote was good enough to share with people.

PI: What made you decide to start writing and believing that people might be interested?

WW: Well, a lot of it came from the feedback that I got from just writing a web log, you know. My *Just a Geek* manuscript is due to O'Reilly on April 2nd and I'm almost done with it right now. While I'm working on it, I have to keep going back into the archives of my website to remind myself of things that happened and to double check dates and facts and that kind of thing. It's kind of cool to have this searchable history of the last few

"The waitress said 'Didn't you used to be an actor?'...I was like, 'Goddammit! Why do I suck?'"

years of my life, but I look at the beginning and it's like, "I'm a shitty writer."

I mean it just--it sucks. It's like, "Woo, woo!" "Have some more passive voice," "he said, looking out the window." You know I mean?

It's awful, and it's been really fun for me to watch my voice um evolve. I'm stunned that people stuck around. That's really interesting. I don't know why people stayed there as long as they did towards the beginning. But, I'm really, really happy that they did, because, without the encouragement, and the support, and the sharing of ideas that has happened as a result of my website, I never would have written any of the stuff.

two thirds of my wife and one third me, because I'm hyper-critical, and I hate everything, and I make sure that I rewrite stuff a lot until I'm real happy with it. If I thought about all the--it's a few million people that read this--I would kind of panic. My heart's beating kind of hard right now just talking about it.

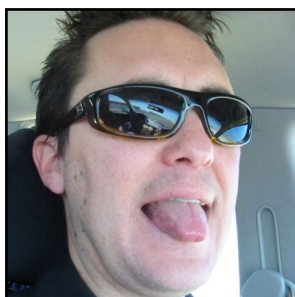
There are a few risks, I suppose. People think that they know me really well. I really only write about 5 to 12 percent of my life; there's a bunch of my life that I don't talk about. [For example,] I like to--this is funny--I really like to smoke cigars. I probably smoke cigars like every other month...I couldn't fucking believe the criticism I got! They were like, "You're promoting smoke and you're the devil!"

"Hey you know what? Fuck you! It's my life. I'm not promoting anything. I'm talking about this thing that I like to do."

And then [there was] this guy who had really, really good, valid, and useful literary criticism about this thing I had written. One of the things he said that really struck me was, "Well this exchange is so far out of character it's unbelievable." I thought, "Wait a minute. But that's... How could it be out of character if it really happened, that I really said those things?" And then I thought, "Well, you know what? There's probably a lot more to the character--not the moral character, but the dramatic character--of my protagonist," who is me in these things. And a lot of people know. And that's kind of exciting for me because I can write all these other stories that really surprise people.

It's funny that I can make an audience excited when I sort of go outside the law, and play an illegal card game, but I can really piss them off when I smoke a cigar (laughs), or talk about having too much Guinness in the pub or whatever. There are all these different facets of me that I talk about, but there are many, many more that I just don't talk about. I mean I gotta keep stuff for myself. (PI: Right.)

You know, If I'm going to write about every single thing that I do, I may as well load up my house with web cams and really get rich.



PI: Well, I've got to tell you, I think that the book was great--a lot of emotion behind it. Why in your writing do you share so many of your personal feelings and your personal thoughts? Is it difficult for you to do that, knowing that you might be setting yourself up, opening yourself up to the people who are reading?

WW: The only thing that sometimes makes me sob is that it can be a little intimidating to think of. If I think about the number of people that read this, I just freeze up, I get performance anxiety, and I can't do anything. Stephen King talks about how you just write for your ideal reader, and my ideal reader is like

“But the idea was just to kind of reclaim myself, reclaim who I was...and just to have, I guess, an ability to speak for myself.”

PI: Right (laughing). Alright, a NetHack question for ya: do you find that you jump for joy when you kill a monster and you know in advance its corpse is edible?

WW: Yea I'm all about [Nethack]. I play Monks, and I'm all about the vegan challenge, so I get really excited. Yeah I get very excited about that.

PI: Is there something disturbing about the number of corpses you have to eat in order to survive the Nethack?

WW: You know, I'm constantly filling myself up until I'm stuffed with food. So, you know, you're having a hard time getting it down.

You know, towards the beginning, I get as many tripe foods as I can, and feed them to my dog, or to my cat. Some dude told me this great trick where you go into a store with your pet, wait for your pet to pick something up, then you walk out of the store and you throw food, and the pet's all, “Hey food!”, and goes and chases it. Then, when the pet eats the food, they drop whatever it is they took out of the store. Then the shop owner won't zap you for stealing it.

PI: (Laughing) The guys are like, “Alright, write this down and make sure that we do that!” I've also read that you're an avid poker player. Does your training as an actor help you bluff your way through the hands and read the other players expressions?

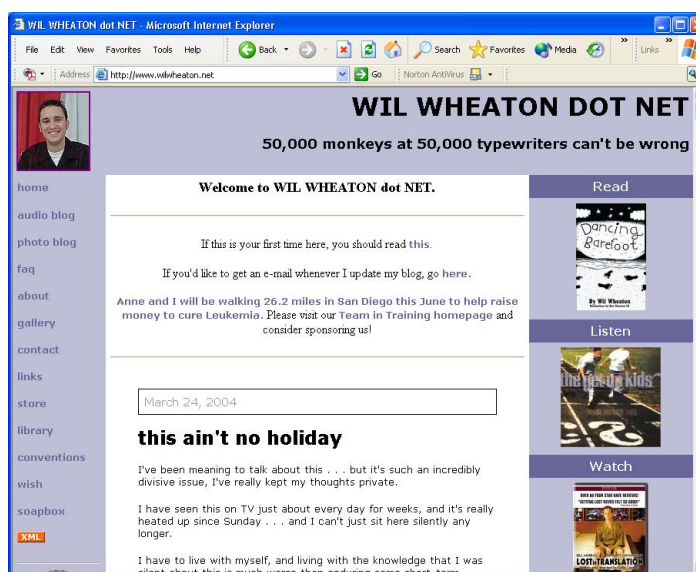
WW: I think so. Every professional poker player who I have read says they don't like to play with actors because I can look down at the hammer, which is a seven two off suit, and I just imagine that I'm looking at a really strong hand, and I can make my heart beat harder, I can make my face flush, I can make myself look excited, and then I just sort of check.

Actually, I don't bluff very often. I really don't like to play hands unless I've really got something. I mostly play low limit and pot limit. I rarely play no limit, so it doesn't make a lot of sense in pot limit games to blow.

PI: William Shatner: current friend or foe?

WW: I don't know. I mean honestly, I don't know. Can we do dead or Canadian instead? (Laughing.)

PI: Sure... (Laughing) Can you give me a little background on why you decided to create Wil Wheaton dot Net?



WW: Yeah, well there's the Hooters story on my website. That's a true story. That is exactly the way that it happened. I should point out that these things happen to me, and if something happens and I think, “Wow! I would never make that up!”—well that goes into the book right away. That goes into the web log right away because I have learned over the last few years that real life is completely different and better than what you make up.

So I went with my best friend Darren to

this Hooters in Old Town, which is ridiculous 'cause I mean, Hooters—it's stupid you know? I mean you may either go to a real sports bar or go to a real strip club. Don't go to something in between.

But we went there anyway and the waitress said “Didn't you used to be an actor?” I was like, “Fuck man... that sucks!” I mean it hurt a lot. I was like, “Goddammit why do I suck?” And people have been after me for years to make a website, to have this online presence. They all wanted to do it as like this big, dumb, commercial, “members only”... they wanted it to be the Melanie Griffith website, and I was like, “You know what? I don't want to do that. I'm not interested in that. I'm already a nerd. I already read other people's websites, and that's just messed up.” That would damage my credibility with myself too much.

So I went to make this Geocities website, and I was like, “You know what? I am an actor, and I'm doing this live sketch comedy show, and it's really good, and I'm going to talk about it, and I bet you fifteen people are going to care, and I probably won't know three of them.” I never thought that it would be what it is today. I never thought it would be what it was a month after I started it.

But the idea was just to kind of reclaim myself, to reclaim who I was (I never thought it would go beyond that—that I wasn't Wesley), and to speak for myself, because I never really had a chance to do that. Sort of to atone for being an idiot when I was a teenager. And to just have, I guess, an ability to speak for myself. That's why I started it.

PI: Ok, if the Internet didn't exist, do you think that you would be able to be as open and honest as you are on your

“If somebody comes up to me to talk...it's not like, ‘Hey, Wesley, I hate you!’ I mean, people just don't do that anymore.”

blog to a live public audience?

WW: (Quickly responding) No. Not at all. I'm really, really shy in real life. Like, I've gotta go see my kids in a track meet today, and I'm dreading it 'cause I don't like to go and sit in the bleachers around these suburban parents who all hate me because I'm ten years younger than them, and my kid is the same age as theirs, and because I'm a Democrat and they're all fucking Republicans.

I feel really uncomfortable and nervous getting out and into public. But when I'm on stage, I can put an imaginary fourth wall between myself and the audience. When I write, I can put a fourth wall between myself and the audience, and I feel ok I feel safe; I feel kind of protected, like I'm sort of behind a one-way mirror.

PI: If you're walking through the mall and you see out of the corner of your eye that someone's approaching you, what goes through your mind? Do you want to run, or you kind of handle it?

WW: I used to want to run. These days, I just sort of wait and see what happens. I think it takes a lot of courage to approach somebody who you perceive as a celebrity, because the risk of somebody being an ass is so great. You know, I don't like to meet my heroes, because I'm worried that they're going to let me down.

PI: So, you think it's better to be known or unknown?

WW: Well, if I'm approaching somebody who's famous, it's better to have a little bit of a reputation, 'cause maybe then I have something to talk to them about.

I was doing some voice work yesterday and the voice actors who were in there, down at Disney, are some of the biggest voice actors in the entire industry. And a couple of them came to me to tell me they had seen my name around town, and they'd seen my website, and they'd heard about my book. I was thrown away by that! I didn't expect that at all. So it's, it's kind of cool to be known.

But one of the happiest times of my life was when I lived in France and I was working on Mr. Stitch 'cause Star Trek wasn't on T.V. in France, and I was completely unknown. I was just "that American that lived upstairs," and it was awesome. I loved it. It was great. It was like I got to be a normal person--I got to be an actor and normal

"The movie's a piece of crap, but I die in it, which is kind of cool"

person at the same time for about four months. It was great.

But if somebody comes up to me to talk to me, as long as they're not an ass, I work really, really hard to stop whatever I'm doing. Within reason, you know, like, "Please don't talk to me when I'm eating," or if I'm obviously having time alone with my wife somewhere. If I'm just shopping or whatever, and somebody comes over to me--I think it takes a lot of courage to do that--it's important for me to stop and respect the risk that this person took to approach me, and then to listen to what they have to say. Most of the time it's something like "I really liked your book," or "I really liked your website," or "I really like *Star Trek*." It's not like, "Hey, Wesley I hate you!" I mean, people just don't do that anymore.

And it's really neat these days when people come up and talk to me about my website; I kind of feel like I'm going to cry because I'm so proud of it. It's so wonderful to be acknowledged for something that I created, that I'm working on right now, instead of something that I did a long time ago, where I was just one small part of the greater picture.

PI: Ok, Sammy Hagar or David Lee Roth?

WW: David Lee Roth, but not David Lee Roth now. (Diamond Dave)



PI: Can you tell us about what you've got going on now? I know you're working on a couple of new projects--*Teen Titans*?

WW: Yep. I'm the voice of Aqualad on *Teen Titans*. I absolutely love it. I've written about this in my website a little bit, but I really figured this out just a few

weeks ago. I was actually in the studio, talking with some of the other actors, and I realized how proud I am of that show, and how great it feels to be totally proud of something that I've done and tell people, "Yeah, watch *Teen Titans*, 'cause it's great. You'll really like it. It's fun. Your kids will love it!" As opposed to, "Oh yeah, I did this fucking piece of shit movie called *Deep Core* cause I needed to pay the attorney bills, because my wife's ex husband keeps taking us to court over nothing." You know, the movie's a piece of crap, but I die in it, which is kind of cool.

It's so great to have something to talk about that I'm so proud of, and to look forward to all the time. I'm kind of doing now what I wanted to be doing through my late twenties. I'm doing enough with myself now that I can stand on the shoulders of the work I did in the past, instead of standing around the work that I did in the past.

PI: The sticker that you have on the Atari joystick that says REPRESENT on your laptop (WW: Yes.)--where'd you get that? We want one!

WW: Listen this is really embarrassing for me. I just want to own up to that right now, alright? I got it at Hot Topic.

PI: (Laughing) Oh, OK.

WW: Every now and then you have to get in touch with your inner poser. And look, I live in suburbia. If I lived in Hollywood, I probably would've seen it in E.K., or I would've seen it at X Large. Someplace else. Maybe Retail Slut. But the fact is I live in suburban Pasadena, and I was at the mall. I was actually--this is even worse--I was at the Yankee-fucking Candle Company buying a Christmas candle with my wife, and I walked right passed Cinnabon and there it was. I saw it in the window at Hot Topic and I said "I've got to have that."

PI: Looks like that's about it. Thanks for taking the time to do this interview.

WW: It's totally my pleasure. I can't believe that that there's interest in this thing I'm doing.

PI: That's pretty cool isn't it?

WW: Totally, totally cool.

Can J2EE and .NET Provide Cross-Platform Interoperability?



This column is the first in a series of monthly articles on J2EE technologies and Microsoft's .NET Framework with emphasis on how to achieve cross-platform interoperability. Based on recent research I realized there were many innovative approaches to achieving interoperability between the two platforms. I decided to focus in the first series of articles on third party interoperability solutions. In the ensuing months I will present business solutions from JNetDirect, JNBridge LLC, and Intrinsyc Software. Interop software solutions are broken down into two categories: software development tools, and plug and play tools. I will also discuss other interop technologies from both J2EE and .NET such as Microsoft's Advanced Remoting technology and Sun Microsystems's approach with SOAP to Web services.

This month's article serves as a brief overview of J2EE and .NET, their architecture and functionality, and then introduces a new, exciting, third party interoperability business solution from JNetDirect Software. Next month, the focus will be on JSQMapper from a technical perspective, a fast and efficient tool for mapping your Java data directly to Microsoft's SQL Server 2000 and earlier releases. Subsequent articles will explore other interop solutions such as JNetDirect's JBIS Integration Server.

Much has been written lately about application integration and interoperability between legacy systems and new emerging technologies and a variety of approaches towards merging the old with the new. The rivalry between Java fans and .NET loyalists always causes endless debates around the issue of market dominance and access to platforms in various computing environments. Such discussions invariably lead to the following question: *if both Java and Microsoft are committed to providing services for clients, wouldn't both parties eventually want to share a common goal and conform to a set of standards that would benefit all developers, albeit Java or .NET?*

Corporations have invested much time and money in developing legacy systems. It seems logical that eventually .NET will migrate towards the open source community and work towards a much needed consistent set of standards. It comes as no surprise that Sun supported an initiative called the Java Community Process (JCP) for the sole purpose of soliciting suggestions from major corporations to further the improvement of all Java platforms including J2ME, J2SE, and J2EE. This opened the doors to third party vendors by allowing them to contribute further to application and cross-platform interoperability and remote access to objects.

Before we delve into application integration, we will first consider platform-independent solutions from the two major players.

J2EE 1.4 Specification

The Java2 Platform, Enterprise Edition (J2EE) v. 1.4, is Sun Microsystems's approach to providing a component-based framework featuring reusable business components. The Java framework contains numerous abstract interfaces that facilitate the flexibility required to generate reusable code and yet support all features found in the Java programming language. This includes the ability to execute applications on any platform supporting the Java Virtual Machine (JVM). Other options include support for transactions, a wide variety of persistence frameworks, and security mechanisms.

Addressing the rapid increase in conducting business transactions on the Internet, Sun recognized the need for supporting XML-based Web services. In a manner consistent with their support for transactions on the Web, Sun frequently releases updates to their Web Service Development Pack (WSDP). The WSDP includes support for Java API for XML-based RPC-(JAX-RPC) 1.1, and SOAP with Attachments API for Java (SAAJ) 1.2. JAX-RPC 1.1, a part of J2EE 1.4, offers the foundation for all interoperability requirements required for supporting Web Services.

The J2EE 1.4 specification designates Internet Inter-Orb Protocol (IIOP) as the primary communication protocol between components, specifically, the RMI-IIOP (Remote Method Invocation Inter-Orb Protocol). Additionally, J2EE uses the Java Naming Directory Interface (JNDI) for access to remote objects.

Microsoft .NET

Microsoft's .NET platform is component based and sits on top of the Windows operating system. The .NET architecture is web services-centric and paves the way for interoperability between technologies residing on the .NET platform. The ability to build applications in any programming language targeting the .NET framework is one of the most exciting innovations released by Microsoft in recent years. They designed the Framework to run on any platform provided the Framework is installed on that system.

The .NET infrastructure presents a multi-tiered approach to distributed programming and uses SOAP as the primary technology for supporting web services. The components consist of classes embedded within namespaces such as System.Object, System.Net, System.Windows.Forms, System.Web, and System.XML.

In a distributed environment, separating .NET's presentation layer from the middle-tier business logic layer is a good practice. The

third tier consists of low-level components for managing services such as database access, connection to legacy applications, and so forth.

The .NET Framework uses HTTP as the communications protocol for messaging and exchanging documents via the Internet. This protocol allows any programming language, platform, or middleware to be an interactive participant in cross-platform interoperability. XML serves as the .NET architectural core component and provides the means with which to transmit XML-based SOAP messages.

The three core components of the Framework consist of the Common Language Runtime (CLR), the Common Type Specification (CTS), and the Common Language Specification (CLS). The CLR performs a wide variety of low-level system tasks, i.e., memory management, error handling, type checking, and security.

The CTS defines common data types all programming languages targeting the Framework must conform to in order to receive multi-language, interoperability support. Remember, a .NET *type* refers to a collection of classes, interfaces, delegates, and structures supported by the CLR.

The CLS sets forth guidelines a .NET-aware compiler must follow before it can receive Framework support. Objects must expose to callers only features common to all .NET hosted languages.

What is Interoperability?

Interoperability, considered in the context of Web services, means all functional characteristics of a web service must remain unchangeable across different operating systems, diverse applica-

tions, programming languages, and data business models. Web services, by definition, must be interoperable. Additionally, no service client must be interdependent upon other services.

Third Party Vendors Provide Interoperable Bi-directional Solutions

Frequently, Java developers find it necessary to map their mission-critical data to databases such as SQL Server. Doing so usually requires an expensive proprietary solution that must be installed on both platforms. Fortunately, JSQL Mapper, a plug and play Java tool recently released by JNetDirect, does the job efficiently. It is easily configured and conforms to the industry-wide XML standard. It allows you to design visually the relational mappings that will be executed by the Java Virtual Machine. JSQL Mapper eliminates the necessity for creating custom code. The best feature is it accomplishes the task without modification to the existing database schema. Not only is this quite a timesaver, but it also eliminates errors and is wonderfully convenient.

The next article will delve more into the inner workings of JSQL Mapper and additionally introduce another interop solution from JNetDirect called JBIS Integration Server. Happy computing and see you then.

Dwight Peltzer is also the author of *J2EE and .NET Interoperability* (McGraw-Hill/Osborne, ISBN 0-07-223054-1). Pick it up at your nearest bookstore today!



ADVERTISEMENT



The Business Software Alliance is the foremost organization dedicated to promoting a safe and legal digital world. BSA is the voice of the world's commercial software industry before governments and in the international marketplace. Its members represent the fastest growing industry in the world. BSA educates consumers on software management and copyright protection, cyber security, trade, e-commerce and other Internet-related issues.



Has Microsoft Finally Gotten the Message?

Terri Wells

At this year's RSA computer security conference, Bill Gates gave some signs that he's heard at least some of the complaints that Microsoft's customers have been screaming for the past few years. Windows, Outlook Express, and Internet Explorer have proven so vulnerable to viruses, worms, and related security problems that it's almost a cliché—not a good thing in a world where script kiddies and more dedicated crackers get their kicks from seeing what kind of electronic mayhem they can cause. So what does the world's most famous chairman propose to do about it?

Well, Service Pack 2 for the Windows XP operating system, which is supposed to come out later this year, addresses some of the issues. First of all, Internet Explorer finally gets a pop-up window blocker. It also gets something called "Windows Security Center," which does a number of things, like give a warning if the PC lacks anti-virus software, or if the anti-virus software it has is out of date or not turned on. (How does it detect these things? Does the software have to come from a list of particular vendors, or is it the type of code that's recognized?) In general, this feature exists to help users keep their PCs more secure.

But that's just one step among many, if Gates can be believed. He would have us understand that he's gotten the message that security isn't a matter of just fixing a few holes and moving on to the next thing. "No single technology can adequately protect against the many different kinds of attacks that computers face," he says in his keynote address. Apparently, this is why Microsoft has teamed up with 10 anti-virus vendors to form the Virus Information Alliance, which offers various educational resources to help users keep their PCs from coming down with nasty bugs—and also with many larger ISPs in the Global Infrastructure Alliance for Internet Safety, which tries to educate and protect users against various electronic attacks.

Education is a good start, but it's only a start. Eventually, Gates hopes to add active protection technologies to the equation. These features, designed to run on a Windows-based network, will adjust the defenses of each computer and feature such nifty things as firewalls that can identify malicious traffic (distinguishing it from regular applications) and stop it, and behavioral blocking that may limit the damage that worms and viruses can cause when they do manage to break through. (Remembering some previous versions of the operating system and related software, one hopes that the company is careful about which behaviors it blocks -- only truly worm- and virus-like behaviors, please, not more normal user-type behaviors! But that's a separate rant).

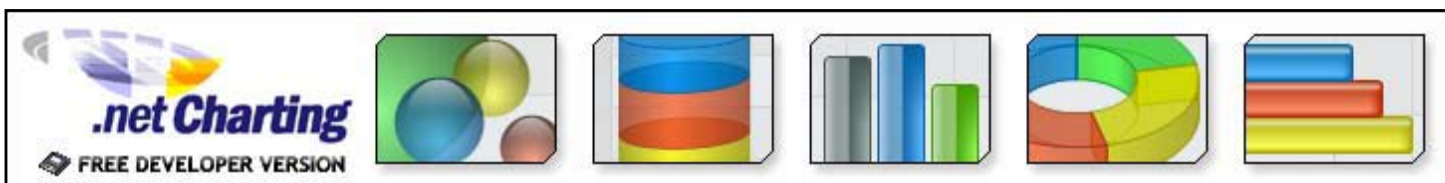
What else does Microsoft's founder hope to accomplish? Well, under the banner of eliminating—or at least significantly reducing—spam, he took the wraps off of Microsoft's Coordinated Spam Reduction Initiative. This initiative is supposed to work by "establishing verifiable identity in e-mail through a caller-ID-like approach, setting reasonable behavior policies for high-volume e-mail senders, and creating viable identification alternatives for smaller-scale e-mail senders." In fact, Microsoft is working on a pilot program for the e-mail "caller-ID" through its Hotmail service.

Am I the only person who gets a little paranoid at the idea of Microsoft administering any kind of e-mail caller-ID system? There has to be a better way.

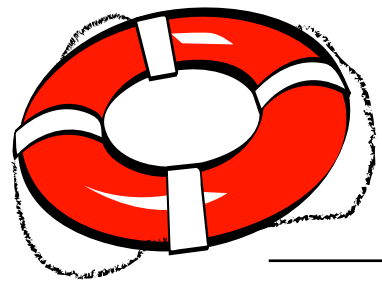
To some extent, there is. It's known as the Real-Time Black Hole List. ISPs can voluntarily sign up for this list; it keeps track of other ISPs that are allowing spammers to use their systems. The ISPs can then choose to not deliver e-mail sent from the offending ISP. If you're an ISP, and you're getting enough complaints from non-spamming users that their e-mail isn't being delivered—and you've got a message from the keeper of the list explaining why—you'd toe the line pretty quickly, right? Needless to say, it doesn't always work, but sometimes, even the threat of being added to the Black Hole List can be enough to encourage an ISP to mend its ways.

Microsoft can be given credit for finally getting the idea that security is important, and it's not something to just add on as an afterthought. These days, plenty of users spend more time dealing with e-mail and going online than they spend watching TV. Since these behaviors can expose users to all sorts of malicious code poised to perform destructive acts on their own or other people's PCs, those who surf and read e-mail need to know how to protect themselves. That is, in part, a matter of education—and any systems administrator should be prepared to tell his or her users the basics, and where to go to learn more. (It's tempting, but you can't blame someone for not knowing.)

But Microsoft has a long way to go before it's clear that Redmond understands the full message. The company's software has proven to be so frequently vulnerable to worms, viruses, and other security breaches that a reporter covering these items could be forgiven for not considering them news. Why should we trust Microsoft with "caller-ID e-mail" when they still have to prove they can deal appropriately with other security issues? The Service Pack for XP looks like it will be an improvement, at least. But Microsoft will have to do better than that before I'm willing to trust them with my e-mail.



OPEN SOURCE: RESCUE ME!



Peter
Lavin

Ever been caught without that rescue disk, and thought that everything was lost, and that you have no choice but to reinstall the operating system? Well don't do it just yet. In this article we'll introduce you to *tomsrbt*, a bootable diskette that will allow you to salvage and/or repair files regardless of your operating system.

Apple has done away with it for years now. My BIOS scornfully refers to it as a *legacy* device. Yes, I'm talking about the poor 3.5-inch floppy. It wasn't always so. There was a time when the 3.5-inch reigned supreme. It was much superior to the 5.25-inch—especially if you remember the 1541 Commodore external drive. I still shudder at the thought. And then there was the cassette tape that you used with the ZX-81. (Now I'm really showing my age!)

In any case, I recently had occasion to be very thankful for this legacy device. My dual boot Win2K/Linux Red Hat 7.1 wouldn't boot into Linux any more; probably something to do with that SCSI card I recently added. "Well," I thought, "it's simple enough to use my Linux boot disk and see if I can sort out the problem." But no such floppy could be found.

Allow me a quick aside to Windows-only readers. Don't change the channel. There is something here for you, because even though the Windows operating system can't view a Linux partition, Linux can see Windows. I'll let you draw your own conclusions about this fact, but one thing this means is that you can use Linux to salvage or repair Windows files.

To get back to our main problem, and to paraphrase an old expression, it looked like I was "up the effluent stream with no means of propulsion." Okay, so I could boot from the installation CD. But in the spirit of not forgetting an old acquaintance, let's see what we can do with only a floppy drive. Besides, as you'll see, an irresistible temptation presented itself to me along the way.

Creating a Bootable Diskette

In the frantic search for my Linux Red Hat 7.1 boot disk, I came across a vaguely remembered diskette labelled "tomsrbt 2.0.103." Could Alice resist the bottle labelled *Drink Me*?

To answer that question you really need to know what *tomsrbt* is (pronounced "Tom's Root Boot"). To quote Tom, *tomsrbt* stands for: "Tom's floppy which has a root file system and is also bootable." I had my boot disk ready-made, but you will have to create yours by going to <http://www.toms.net/rb/>.

Locate the link for downloading, find an appropriate mirror and choose the file named "tomsrbt-2.0.103.dos.zip" if you wish to create your diskette under Windows, or "tomsrbt-2.0.103.tar.gz" for Linux/Unix.

A word of warning at this point: if you are running Windows 2000, creating this diskette is a bit problematic. If you've got a Windows 98 machine handy this provides your easiest solution. Copy the zipped file to this machine, decompress it and boot into DOS to create your bootable floppy. Otherwise see, <http://not.toms.net/wiki/bin/view/Tomsrbt/CreateTomsrbtFromWin2000>. Various solutions are presented there but the next easiest seems to be to make DOS boot disks using the resources at <http://www.bootdisk.com> and then create your Linux diskette.

A couple of final notes: This distribution of Linux does not support writing to the NTFS file system, but check Tom's site for any updates. There should be no problems creating your bootable disk under Linux.

Booting From a Legacy Device

Let's assume that you have managed to create your Linux boot diskette. Reboot using this diskette, first making sure your BIOS is set to boot from the floppy drive.

Okay, I know it's taking a long time to load the operating system but go ahead and use the time to reflect on that old acquaintance—and be comforted that you are not using a 5.25" drive.

Choose your language - yes you get to choose your language if you don't want the default; pretty soon you'll be looking at a screen full of all the Linux commands available. If you're a little shaky at the command line in Linux or you are one of the fearless Windows-only users who has followed us this far don't worry. The operating system commands you need are limited in number and we're going to explain them as the need arises.

At this point you can remove the diskette from the drive. You don't need it any more because you are now running from a RAM drive. Besides, you may want to format a diskette (Linux or Windows) and copy files to this drive.

That blinking cursor wants you to log in. Your only choice is to enter `root` and after that the password, `xxxx`.

"ls" and "cd" Commands

The most basic command you'll need is `ls`, the Linux equivalent of the DOS `dir`. Try it out and you should see a list of directories (folders in Windows terminology). If you want to see more information about these directories then type `ls -l`. The `-l` is a switch that gives the "long" view of your files.

Next, you will want to navigate around the file system and for this you'll need `cd`. Looks familiar doesn't it? In fact, it works exactly the same way as it does under DOS. Use `cd` with a directory name and that directory will become your working directory. Returning to the parent directory is as simple as `cd ..` or to go to the root directory `cd /`. Don't forget the space, or it won't work.

Mounting a Drive

By now you are probably itching to look at your Windows file system especially if you've never seen it from Linux before. To do this, the file system needs to be mounted, and in order to be mounted it needs a mount point. What this boils down to is that you need to create a directory and mount a specific file system on this directory. It is conventional to create mount points in the `mnt` directory. If you are at the root directory and issue the command `ls` you should see this directory. Make it your working directory by typing `cd mnt`. You'll find that there are no subdirectories

here but we are going to make one. Again the command is identical to the DOS command. Type `mkdir cdrive` and press **Enter**. Now list the contents of the current directory to see the newly created directory. There you go. You've created a directory that will be the mount point for the Windows file system.

NOTE Remember though that you are working with a RAM drive so the changes you are making and directories you are creating won't be there the next time you boot from the diskette.

At this point you need to know something about the way that your hard drive is organized. If you are only running Windows and have not partitioned your drive then your C: drive will be referenced from Linux as `hda1` and the command to mount this file system is `mount /dev/hda1 /mnt/cdrive -t msdos`. Try it. You'll only see a message if you've made an error. Now change directories to the `/mnt/cdrive` directory. Not sure where you are? Determine your present working directory by typing `pwd`. To look at your Windows files you simply need to change to the `/mnt/cdrive` directory and use the `ls` command. Savour the moment if this is the first time that you have seen a Windows partition from Linux.

If you are running Linux then I'll assume you know which drive to mount. Again, if you only have one partition then it will be `hda1`.

You should always unmount a drive when you are finished. Your C: drive would be unmounted by typing `umount /dev/hda1`. Make sure the drive you are trying to unmount is not your present working directory or you'll get a "drive busy" error.

Copying Files

If you're here to salvage some files by copying them to a diskette then you will also need to know how to mount and format a floppy. You'll be able to format your diskette for DOS or for Linux by first typing `fdformat /dev/fd0H1440`. To make a Linux floppy type then `type, mke2fs /dev/fd0` or, alternately, `mkdosfs /dev/fd0` for a DOS diskette. To copy files to your floppy you will again have to mount the drive. This can be done in exactly the same way that any other drive is mounted. First create a mount point, let's create one called "floppy", and then mount the drive by typing `mount -t msdos /dev/fd0 /mnt/floppy`. Just drop the file type switch (`-t msdos`) if you are mounting a Linux floppy. We're now ready to copy files.

Copying files is identical regardless of the operating system. You simply use the `cp` command specifying a source and destination. For instance, to copy the file `fstab` in the `etc` directory to your floppy you would simply type `cp /etc/fstab /mnt/floppy/fstab`. If you wish to copy a directory and all its contents then you will need to know the `-r` switch. The "r" stands for recursive and will copy all files and directories within the directory specified.

In this way you can salvage as many files as needed from the hard drive. Read on for more advanced commands.

Advanced Commands

In some cases you may want to run as root on your original Linux partition. This can be done by creating a mount point for the file system, mounting it and then using the `chroot` command. This is exactly what I wanted to do so that I could make a boot disk for my Red Hat 7.1 installation.

First I created a mount point for the file system. In my case I knew that my Linux drive was device `hda5` so I created a mount point called "rh71", mounted this device and then simply typed the command `chroot /mnt/rh71`. I next changed my directory to the `sbin` directory, typed `./mkbootdisk --device /dev/fd0 2.4.2-2` and inserted a floppy into the drive. The number following the device is the kernel version number of my distribution. You don't have to remember it—you can easily look it up by examining the files in the "boot" directory.

When I was finished I changed back to `tomsrtbt` by typing `exit` at the command line.

In very quick order, I had a boot disk for my Red Hat installation. I could now boot into Linux and do anything else I needed to do using X Windows.

Things don't always work out so nicely though. Sometimes you may need to look at or change information in a file. In this case you will have to use the text editor `vi`. I'll give you some quick tips on using this editor.

`vi` is invoked by typing `vi` and pressing **Enter**. If there is a specific file that you wish to edit type the name of the file after the command, `vi`. Upon start-up, `vi` is in command mode and you need to press `i` for "insert" if you wish to add to or change the file. Saving your changes is done by returning to command mode by hitting the **Esc** key, followed by `:wq`. This is not an extended tutorial on using `vi`, so if you have made mistakes and want to abandon what you've done you may quickly exit by pressing the escape key and then `:q!`. None of your changes will have been saved.

There are many more useful commands available—for checking the integrity of the file system, for instance—but you would probably do well to learn a little more about Linux before exploring them. You don't want to accidentally reformat your drive!

Conclusion

The "tomsrtbt" distribution is a nice *little* tool if you have trouble booting from your hard drive and don't have a rescue disk. Using a few basic Linux commands, files can be examined, copied or changed regardless of what operating system you are running.

Peter Lavin runs a Web Design/Development firm in Toronto, Canada. For more information visit <http://www.softcoded.com/>. He may be reached at peterlavin@sympatico.ca.



MySQL vs. Postgresql

VI vs. Emacs

Perl vs. PHP

Postfix vs. Qmail

Paper vs. Plastic

It doesn't matter
which side of the
holy war
you choose.

You're in
control here.

VPS
Hosting
Starting at
\$14.95/month
Plus FREE Domain
globalservers

Questions?

US&Canada

1866-HOSTED-1

International

801-437-6161

SCO v. IBM: The Story Until Now

Terri Wells



About a year ago, SCO (then Caldera) filed a suit against IBM. SCO wanted \$5 billion from IBM because IBM disclosed SCO's confidential technical information, or so it claims. This alleged breach of confidentiality supposedly occurred when IBM changed its business strategy to focus its efforts on Linux rather than IBM's own versions of Unix, AIX. SCO and IBM had, at one point, been working together on IBM's AIX. SCO says that code from its System V operating system somehow made it into Linux, and that it's IBM's fault that this intellectual property is now public knowledge.

IBM, being nobody's fool, countersued in August, accusing SCO of patent infringement. Here's a little tip: if IBM sues you for patent infringement, be afraid, be very afraid. When it comes to companies with patents, IBM is the world leader, hands-down. Microsoft, Sun, Intel, Hewlett-Packard? Not even close.

Of course, once it becomes an issue of who owns what and who is entitled to what, Novell had to get involved. But wait, didn't Novell sell System V to SCO/Caldera way back in 1995? Sort of. If you check out Novell's Web site, you'll find that they still own the copyright on Unix System V releases from release 2 all the way to release 4.2MP. And, to judge from the hot-and-heavy back-and-forth letters between Novell and SCO, Novell does not believe that it sold to SCO the right to harass people to buy a Unix license when they're using Linux.

It's possible I've quoted this before, but here's the relevant passage from a three page letter, dated May 28, 2003, from Novell to SCO: "Importantly, and contrary to SCO's assertions, SCO is not the owner of the UNIX copyrights. Not only would a quick check of U.S. Copyright Office records reveal this fact, but a review of the asset transfer agreement between Novell and SCO confirms it. To Novell's knowledge, the 1995 agreement governing SCO's purchase of UNIX from Novell does not convey to SCO the associated copyrights. We believe it unlikely that SCO can demonstrate that it has any ownership interest whatsoever in those copyrights. Apparently, you share this view, since over the last few months you have repeatedly asked Novell to transfer the copyrights to SCO, requests that Novell has rejected." Of course, SCO disputes this.

And it looks like District Judge Brooke Wells is the one who's

going to have to sort out this mess. Just this month she ordered both SCO and IBM to put up their code. Specifically, "SCO is to provide and identify all specific lines of code that IBM is alleged to have contributed to Linux from either AIX or Dynix." SCO is also to identify all the lines of code in Linux that it claims rights to. IBM also has to provide releases of AIX and Dynix to the court in support of its case. Furthermore, IBM was told to pare its 7000+ long list of witnesses down to 1000.

Will SCO be able to chase after everyone using an open source version of Unix and make them pony up for a license? With SCO's recent suits against DaimlerChrysler and AutoZone, you might think so—but don't bet on it. Both Hewlett-Packard and IBM recently closed deals with Novell relating to SuSE Linux, and you wouldn't see big companies getting behind something like that if they were worried about lawsuits cramping their style. HP's deal expands on an earlier agreement, enabling it to support Novell SuSE Linux "from the desktop to the datacenter," as they put it, since this new deal is specifically for desktops and notebooks. "[W]ith certain support contracts, HP will provide qualified customers indemnity against possible litigation from SCO." That doesn't sound like a scared company to me.

IBM's agreement with Novell lets IBM ship SuSE Linux Enterprise Server preloaded across its entire server line. It's a sweet deal, but not unexpected, given the support that IBM has given to the open source movement. At the same time that this deal was announced, the two companies also revealed that IBM was investing in Novell. IBM is buying Novell Series B convertible preferred shares that are convertible into eight million shares of Novell common stock at a price of \$6.25 per share. These Series B shares, by the way, get a dividend of two percent a year, payable quarterly in cash. This agreement was made in connection with an earlier IBM announcement that it was investing \$50 million in Novell. Again, these are not the moves of a company that's afraid that open source software is going to become "too hot to handle."

I find it hard to believe that SCO has much chance of winning this case. Intellectual property law is pretty dynamic right now, and still somewhat murky when it comes to software, so maybe SCO thinks it has a chance. It will be interesting to see how this finally plays out; it might even set some precedents.

Savvy developers use Elance to profitable Outsource projects



on line

Gain Free Access to Qualified Professionals

What do you need?

- ✓ If you are looking for specific expertise, search profiles of available professionals.
- ✓ If you have a defined project, post a project description and get competitive quotes.
- ✓ Not sure how Elance can help? Browse service packages to find a deliverable.

Web Design and Development

Find a service provider to do your project

[View ALL Service Providers](#)



Captain Media

Feedback: 4.5

Complete Web Solutions

[View Profile/ Buy Services](#)



netsoftglobal

Feedback: 4.8

Premier Web Solutions from Nets...

[View Profile/ Buy Services](#)



nuramedia

Feedback: 4.8

Premier Web Solutions from



View projects posted by others

[View ALL Providers](#)

Project Name	Bids
Personal/Dating backend development	10+
Community Portal	10+
develop web based healthcare10+ salary survey	
Web Database from Pagemaker Files10+	
Software Development	10+

HARDWARE:

x86-64: The Golden Handcuffs



With the discussion over x86-64 becoming prevalent now that Intel has decided to adopt it for their Xeon workstation processors, and likely, their desktop equivalents at some point in the future, I figured that it was a good time to go over what this means to the hardware and development communities. We'll be taking an in depth look at how those pieces of silicon work.

It would be very difficult to explain how Intel's move truly affects anything without a look back at what an instruction set architecture really means. So that's where we'll begin.

Instruction Set Architecture

In the language of computers, the "instruction set" is all the possible words in the dictionary. Of course, to a computer these are all represented by 1's and 0's, but we'll pretend that they're letters and numbers.

Instructions are the lowest level that can be coded at. In comparison, today's code, written to be executed in Java or C, is considered **high level**, as the programmer is not aware of the hardware that this code is being run on. Before 80x86 came along, most programmers were content to write at this base (or low) level. Writing code at the **low level** means you are moving data along the paths from register to register, and to the functional units of the processor.

Registers are the smallest parts that are "user accessible," and are nothing more than a really small data storage space. Data must be moved into registers for it to be worked on, since you can't actually perform an operation on something that is in any of the parts of the storage hierarchy,

be it cache, ram, or hard disk.

With many instruction sets, hand coding at this level is not too difficult, and ends up being quite efficient. The types of instructions available allow for most math expressions, such as adds, subtractions, multiplies, and divides, as well as control commands such as loads and stores to and from memory, shifting bits within registers, conditional and unconditional branches, and jumps to different places in the program.

The specific instruction set determines what type of machine the processor is. There are two main overall design types, CISC (Complex Instruction Set Computer) and RISC (Reduced Instruction Set Computer).

CISC vs RISC

Right now, the only mass-produced CISC processors are those that dominate the desktop market, the x86 processors from AMD and Intel. Since the 80's, many newer RISC processors have tried unsuccessfully to knock CISC off the map entirely, and been completely unsuccessful in doing so in the mainstream desktop market. While the lines are starting to get blurred between the types, I'll do my best to explain the difference between them.

CISC came along first, and was meant to take the complexity away from the compilers (which weren't very good back in the 60s-70s) and put it in the hardware. This reduced the lines of code, as actions like multiplication or addition, which needed operands from memory, took only one instruction. Additionally, CISC machines typically have a small number of specific purpose registers. Since the number of variables in code is usually higher than 3 or 4 at one time, this induced a lot of what is called register spilling. With a smaller amount of area to hold these variables at one time, some must be moved in and out of memory (which is a permanent place for holding data).

Newer CISC designs make up for register spilling by having more "hidden" registers, into which values can be moved temporarily by the hardware. However, you can't use them from the programmer's model. CISC architecture is typically known to be a "register-memory" or "accumulator" type. This means that in most cases, operands are capable of being specified from memory, as well as the registers. This adds to the complexity of the instructions, which first have to move the data from memory to the register, then execute, and finally update whatever values were worked on. Complex instructions mean complex hardware. Even though transistor size allows for large numbers of instructions, the metal wires which interconnect units--as well as move data around, relay the control signals, and complete other communications necessary to run the CPU--are starting to get overloaded.

Web Developer Certificate

Choose 19 or 20 days of hands on training for \$5,900

[Click here to Learn More](#)



RISC has been developed for over 20 years now. Unlike CISC, it moves the complexity to the compiler. Today, compilers are capable of making better low level code than humans, tailoring specifically for more complex architectures, and easily switching between different designs. This allows the processor design to be much simpler.

These “load/store” architectures typically allow for operations to only be executed on registers, meaning that you have to explicitly bring in each piece of information to the register file. What might be one instruction on a CISC type implementation might take 4 on a RISC machine (load one operand, load the second, execute, and store back to memory), increasing code size. This might seem slower, but four instructions don’t necessarily mean slower performance than one instruction. Remember, all those same things have to be done in the CISC instruction, they just are just done by the hardware on its own. By removing this complexity, there is more die space left for other uses of transistors.

Typically, RISC style machines have many more “general purpose” registers (GPRs) available to the compiler. Older designs of MIPS (Stanford) and SPARC (Sun) for example, as well as PowerPC (IBM) and Alpha (DEC) had 32 GPRs. Newer ones have even more; x86-32 machines, such as your current Athlon XPs and Intel P4s, by comparison, have eight GPRs. Itanium (IA-64) has a massive 128 GPRs available to the compiler. By having more GPRs, there is less of a need for load/store commands to slow (relatively) memory, especially in looped code.

As I said, the lines are blurring. x86 based processors have become more like RISC. With the introduction of 80386, eight of the specific use registers from 80286 became “general purpose,” in addition to being extended from 16 to 32 bits. Some RISC designs have had CISC type instructions added to their ISAs in order to fill specific needs in various multimedia applications.

There are still some key differences. Most specifically, x86 still has many different sizes for each type of instruction. There is no specific length, depending on the extensions needed to choose from operation size (8, 16, 32, and now 64 bits), regardless of whether its meant for the MMX, SSE, x87, or normal execution units, what type of instruction it is, and so on. In most RISC designs, the instructions are all one length, again taking away some complexity.

x86-32, IA-64, And Now x86-64

As I mentioned before, x86 is considered a CISC type design. Having experience with much simpler RISC architectures, as well as more organized CISC ones like the Motorola 6800, I

must admit that Intel’s design is very intimidating. It’s also cursed with having to be backward compatible for code from late 70s. As noted in a fantastic lecture at Stanford by one of Intel’s former Chief Architects (<http://stanford-online.stanford.edu/courses/ee380/040218-ee380-100.aspx>), it’s a pain to carry on old parts of the design you know no one uses. However, taking a gamble with a mainstream processor—on the odd chance someone who makes a popular program suddenly uses that old instruction—is not worth the consequences.

Almost 10 years ago, Intel and Hewlett Packard (HP) caught on to the fact that it was time to start from scratch, and go another direction. What came out of this was Itanium. It was meant to be a 64-bit replacement for high end computing, and eventually work its way downwards to be a complete line replacement for x86.

Itanium, in its ISA design, fixes many of the shortcomings of x86 that have come to light in the last 30 years. Intel, it seems, made a mistake in designing for the high end, as it seems at this point impossible to shift Itanium (some call it the *Itanic*) to a more friendly cost point for those that don’t need to study weather patterns or tectonic plate shifts in the Earth’s crust.

Additionally, Intel mistook the desire of consumers to continue to use old code on their new machines. As Mr. Colwell mentioned in the lecture, people are weird. The general public is very attached to the programs they are used to, and steadfastly refuse to move on to something new (regardless of how good it might be).

Along came AMD, needing a gimmick to make up for another Joe Six-pack’s “MHz RULE!” myth. Since Intel has decided to devalue MHz, and march along at ever increasing speed regardless of what that does to efficiency, AMD needed a “blue crystal” to sell their product: 64-bit. So they went on and extended x86’s registers, much like what has happened before going from 16 to 32, and 8 to 16 before that. And in the process they have extended our suffering with the “golden handcuffs” of x86 for another 10 years.

In the workstation market, and the server one as well, 64-bit is certainly needed. The reason? 32-bit addresses only allow for up to 4GB of memory. To those of you who now think that 1GB is the least you’ll buy for a new computer at home, you should keep in mind that this was surpassed long ago by systems doing CAD work, simulation, and other development. The fact of the matter is that any machine based off of Xeon or Athlon MPs had to use hacks to get around the 4GB limit for the system. Even then, 4GB was the limit that could be donated to a single program. In a large database for example,

want to be FAMOUS?

Only by accepting articles from authors involved directly with the technology they write about can we properly cover the multi-faceted world of technology. That’s why Developer Shed is looking for a few good writers to join our team. We’re looking for tutorials in all areas of technologies and of all technical levels. We’re also looking for writers who can formulate and defend an opinion, review new software, hardware, and services, and cover the latest news about the business of technology and technological advancements. Real world stories of how you use technology in a creative or unusual way, and/or how you’ve formulated an interesting solution to a common problem are also highly sought. Mostly, we’re looking for experts in their field who want to share their knowledge with the world-wide community of developers, programmers, and engineers which Developer Shed is focused on serving.

To join the Developer Shed team of writers, you must first fill out a writer request form, found at <http://www.developershed.com/writers.php>. Please include a brief description of your background and areas of expertise, including particular programming languages, tools, and hardware. Also include what ideas for articles you have, and one (1) original writing sample for publication in any of the Developer Shed sites. The article cannot have previously been published by any other website and must pertain to the world of technology today, following the aforementioned topics (first paragraph). The article must be 1500 words or longer, subdivided into at least four (4) pages, and submitted in RTF, HTML, DOC, or SWX format. You will then be contacted by a member of our editorial staff who will communicate any requirements at the time (such as topic suggestions and time-tables). Note that your writing sample will be used for publication and that all articles submitted are done so gratis in order for our editors to see the accuracy, detail, and overall composition of the article, in addition to audience response.

If you qualify, at that point, we will offer you a contract for writing with us. All articles you write from that point on that are acceptable for publication will be paid at the rate of \$50 (USD) per article. In addition, you will be able to benefit greatly from what we have to offer outside of that. We give our authors a large amount of publicity by linking to their stories all over our network, linking to their home page, giving them an author bio, and featuring their articles in things such as our newsletter (which gets sent to over 100,000 people on a weekly basis). In fact, if you are the author of a book, we’ll also help you give the book some publicity. Considering we have more than 300,000 individual readers on a daily basis, and hold a Google ranking of 8 amongst most of the websites in our network, that carries with it a lot of weight. In short, you’d benefit heavily from publishing with us.

To get started, go to <http://www.developershed.com/writers.php>.

that's still not sufficient.

The solution? Make a move to 64-bit addressing and pointers. Sun, IBM and others made this move a long time ago. That allows RAM addressing in the level of terabytes, for both the system as a whole, and specific programs.

"But isn't 64 twice as fast as 32?" Not quite. Fielding questions like this make me wonder how Joe Six-pack can invest thousands of dollars into something, and not have a clue how it works. Looking at it from a current standpoint, for **floating point** (FP) operations, which can make use of the extra precision, we already have 64-bit. As you probably know, FP numbers are numbers with a decimal in them. Computers are by design inaccurate working with these types of numbers, since they can't hold fractions. The more precision (more bits) you give them, the less of a problem "rounding errors" become.

FP was not originally in the x86 architecture (which gives another indication of its age), and was added later as a separate co-processor with another instruction set called x87. Since that time, they have been added on die, and are now an integrated part of any x86 based architecture.

x87 is 80 bits wide and stack based. It also suffers from terrible performance relative to other FP implementations. SSE (Streaming SIMD [Single Instruction Multiple Data] Extensions) and SSE2 were added later to bump up multimedia application performance. As a bonus, they also do a much better job of FP operations, both performance and precision wise. SSE2 is already 128 bits wide, as shown in figure 1, and it holds two 64 bit floating point values, or 4 32 bit ones per register. So going to a full 64 bit architecture gains you nothing that way.



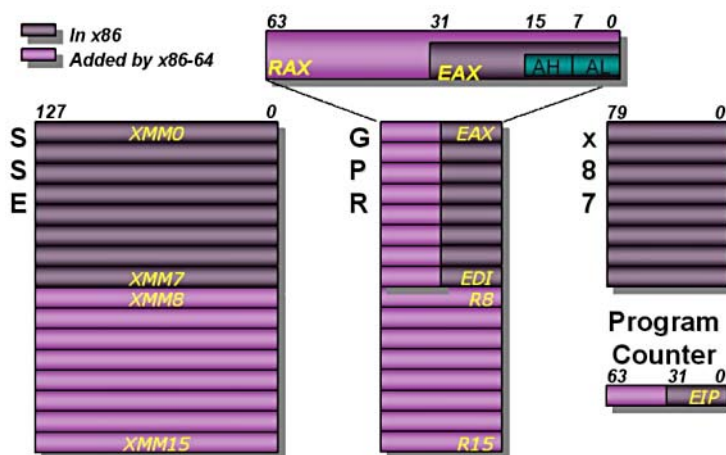
Anything that fits into 16 bytes

Figure 1 SSE2

Lately, both AMD and Intel have encouraged compilers and coders to make use of SSE2 instead of x87 for the obvious reasons. But what about for integer math? At this point, most operations fit nicely into 32 bits. There will of course be some advantages that will be gained automatically: previous operations that had to do two operations and save across two spaces to take care of a value that does not fit into 32 bits (any number greater than 4,294,967,295) will now be able to be done in one shot. For now, those are rare, but to quote *Field of Dreams*, "build it, and they will come." Eventually there will be commercial programs to exploit this. Just not now.

What AMD did with their upgrade to x86, is add more visible registers. As you can see in the next picture, in addition to extending each register to 64 bits, they also doubled the number of SSE and GPRs.

Figure 2 AMD's SSEs and GPRs



As I said before, having more registers available to the compiler can help lower the number of load/store calls, which are slow. Less of that kind of instruction also brings down the number of lines of code. However, because there is need for an extra byte in each instruction to define the 64 bit instructions (the RAX extension), optimized code ends up being slightly larger, because of the extra space required for 64 bit values. According to AMD, code size is up on average about 5%, while the number of instructions is down by 15% when recompiled for x86-64. With larger cache sizes being found on processors these days, as well as cheap storage, this isn't much of an issue.

Conclusion

I hope this article helped you understand a little more of the background behind the move made by both AMD and Intel to x86-64. It's really not much of an advantage over x86-32, other than the gain in memory addressing, and the extra registers which make the last mainstream CISC architecture more RISC like. I'd prefer to see the extermination of this well worn architecture, and see it replaced by something more in line with what we know today. Starting fresh would be a much better option performance-wise than constantly trying to deal with improving an archaic design such as x86 without breaking compatibility. Unfortunately, some of the more outstanding issues have been dealt with, and the life of this instruction set has been extended yet again. Somehow I have a feeling in 10 years I'll be writing a similar article on how x86-64 moved to x86-128. But we'll find out in 10 years.

D'Arcy "DMOS" Lemay is a 4th year electrical engineering student at Queen's University. He's specialized in processor architecture and power applications. Mostly, this has been a degree in alcohol consumption."



ROBOT RACERS: the Agony of Defeat



NASCAR it wasn't. But, for those interested in what autonomous vehicles can do, the special race recently run in California's Mojave desert was an eye-opening event. The race was dubbed *The Grand Challenge* and sponsored by the Pentagon's Defense Advanced Research Projects Agency (DARPA).

by Terri Wells

Fifteen teams were cleared to compete during the week-long qualification, inspection, and demonstration period just before the race. During the months leading up to the race, they were required to submit technical papers about their entries. These weren't remote-controlled vehicles; they had to be able to navigate the course and avoid obstacles by themselves. To do this, many used various kinds of sensors (such as infrared), as well as GPS devices. To win the race, a vehicle had to navigate the 140-plus mile course (from Barstow, CA, to Primm, NV) in the quickest time within a specific 10-hour period. The teams competed for pride, glory--and a million-dollar prize purse.

Sadly, not a one completed the course.

Of the fifteen vehicles that were cleared to compete, only thirteen actually started the race--two, including a riderless motorcycle, pulled out before

then. Of the rest, an assortment of Humvees, pickup trucks, and dune buggies, roughly half didn't even make it further than a mile; many of those didn't make it out of the starting area. Only four vehicles made it more than five miles. The most promising one, a modified Humvee fielded by the Red Team, drove nearly seven and a half miles before it went off course, caught itself on an obstacle, and had one of its front wheels catch fire. (Talk about a bad day!)

Most of the other entries experienced other difficulties: throttle problems, unexpected backing up, barbed wire, fuel leaks, brakes locking up, not finding the route, and simply getting stuck. And these folks weren't a bunch of amateurs. Granted, one team was from Palos Verdes High School (and no one would have cheered harder than me for them to win, except maybe their parents), but the stringent rules and qualifications run to 28 pages--all but guaranteed to keep out the casual.

For example, the Red Team was made up of Carnegie Mellon graduate and undergraduate students led by Professor William "Red" Whittaker, an apparently well-loved and respected teacher with a passion for "autonomous robotics in harsh real world (and extraterrestrial) environments." CMU has a tradition of techno-racing dating back to 1920. The Palos Verdes Road Warriors (the high school team) included student members with extensive computing experience, even C/C++ programmers--and one of their members placed first in the L.A. County Science Fair Robotics Division for an autonomous vehicle. SciAutonics II, whose vehicle made it nearly seven miles before going into an embankment and getting stuck, was made up of Rockwell Scientific employees and external consultants. Team Digital Audio Drive featured a pair of brothers who are robotics enthusiasts. Axion Racing included business professionals, scientists, and specialists, and boasted two sponsors. Team TerraMax sprang from a company named Oshkosh that develops hybrid propulsion systems for tactical defense trucks. These teams, a representative cross-section of the kinds of groups competing, put nine months or more of serious effort into this challenge.

So why didn't the vehicles get any further than they did? Well, this race was probably the first of its kind, and the race course offered up extremely difficult terrain. DARPA was actually pleased that some vehicles managed to get as far as they did. The whole

point of the Grand Challenge was to help spur research on autonomous vehicles, so it's a little early to expect miracles. The US military wants to start using these unmanned vehicles for missions where one wouldn't want to send soldiers, because they're too dangerous (like clearing a mine field), or too dirty, or don't really require a large amount of intelligence.

What else can you do with an autonomous vehicle? Well, exploring the surface of a distant moon or planet is one application that comes to mind. These vehicles can also be used for similar applications here on Earth. More "down-to-earth" applications include unmanned tractors that harvest crops by themselves and "driverless" cars that use their sensors to successfully navigate rush hour traffic.

But why talk about the Grand Challenge here? Can we really expect to see some trickle-down technology? I can already see those of you who know DARPA's background snickering a little at these questions. No doubt you're remembering a project of theirs that had a huge trickle-down effect. Back when that project was started, it was called ARPANET, and it connected computers in such a way that researchers could more easily communicate with each other electronically. The purpose behind the original project was to safeguard against the possible collapse of the regular communications network (and therefore chain of command) in times of war. Sound familiar? It should; it led to e-mail, the Internet, and the World Wide Web.

I don't think I need to reiterate the history behind ARPANET here; just put the name into the Google search engine and you'll find a ton of links that will take you to that information.

Which, in a sense, makes my point for me; they planted a seed, and it grew into all sorts of wondrous things that could hardly have been foreseen at the beginning of the project.

DARPA is approaching this latest project in the same spirit. The FAQ for the Grand Challenge included the question "What will a Challenge vehicle look like?" The answer? "The Grand Challenge is designed to establish a desired level of autonomous performance without specifying how that performance is achieved... who knows what could slither or crawl across the starting line."

This is not to say that we'll see driverless vehicles and the end of road rage in the next five years, or even the next ten years. But we can certainly expect some real excitement in that area in the future. Start your engines!



Freelance writer, Terri Wells, has spent five years covering the Internet and other high technology topics. She currently spends her days working with technophobic professors at Duke University.

The Developer Shed Weekly Newsletter

GET IT! (GET IT?)

Staying up-to-date isn't always easy. That's why The Developer Shed Weekly Newsletter is the right choice for you! All the great information from our line of sites, delivered right to your mailbox in one weekly package. Best of all: it's FREE!

SIGN UP TODAY! (GET IT?)



Advertising in



plug



MAGAZINE

just makes Gooood Sense!

The Developer Shed Network has an established reputation in the technical information community by continually offering our readers a one-stop-shop for information on both well established and emerging technologies. Plug In reaps many of the benefits offered by the Developer Shed Network's popularity with exposure available to it via our sites and publications.

- **Over 10 million unique visitors to our network every month.**
- **Over 150,000 opt-in subscribers to our weekly newsletter.**
- **90% of audience will base purchasing decisions on content provided throughout our network.**

We go the extra mile to make your ad dollars work efficiently and effectively. That's why advertising with us just makes Gooood Sense!

SITE PROMOTION: ON RIGHT SIDE OF THE LAW: Keeping Your Web Promotion Ethical



Trying to stay on the right side of the law in terms of web promotion can be extremely difficult at times. The rewards of being found at the very top of the search engines can sometimes force Search Engineers and web designers alike to step over the mark using dishonest methods and both spoil the search engine results, and ultimately have their own sites penalized.

Being unethical is something that almost every Search Engine Optimizer has been guilty of at least once, and for some Optimizers, for most of the time. Methods used have changed over the years from same colored text as backgrounds, to doorway pages to IP filtering. Each has served a purpose, and some still do, but eventually, using only these methods and not laying foundations with good old-fashioned SEO will be the death of most sites.

To combat unethical web sites, search engines have updated their methods of finding and eradicating the sites that are guilty of cheating the system. As many people will know, the recent updates from Google, most noticeably the Florida update, have not only changed the way that Google uses commercial web sites, but also helps in a large way to rid the web of many sites that were abusing the system.

This is of course excellent news for the searcher, who can now find what he wants much more easily, but what of the clients of SEO companies whose websites have disappeared? What do they do now? Companies that have enquired about our own services since these recent Google updates have been alarmed at the tricks used by previous SEO firms on their own sites that ultimately got them banned or punished.

These are the top 5 basic rules of achieving an ethical website that will never be punished on search engines, no matter what algorithmic changes may happen.

• Build a site FULL OF Content

Look to build a minimum of 50 pages of relevant content for the site. Of course, the more pages of content, the better. Write articles about your products or services. Pages like 'About Us'

and 'Contact Us' usually do not contain relevant content and you should not count them as such.

• Keep it Simple

Keep the web pages as simple as possible. Try to use HTML 3.2 rather than 4. Text in your website should outweigh the HTML code within. Keep JavaScript away in '.js' files if necessary to include in the pages.

• Add Content

It is important to add content on a regular basis to the site. This will help the engines to see that your site is constantly updating. This will increase importance over competing sites who haven't updated pages since 2002 for example.

• Cross Linking

It is very important that search engine spiders can find all of the pages that you want it to see. To do this, make sure you have good links, preferably text links, to most pages all across the site. A simple site map is also a must.

• General SEO

A very generic rule, I know. Make sure on each page that your head tag is simple and does not contain any tags that are not needed. Make sure that you include the main keyword for that page in the title, the description, the keyword list, the H1 tag (if possible), the body text and link text.

Using these 5 rules as a start will help to outperform any website that uses mainly tricks to obtain good positions on the search engines over a period of time. It may take 2 weeks or it may take 12 months, but eventually you will beat the cheats at their own game by simply being ethical. At some point they will be punished or eradicated altogether.

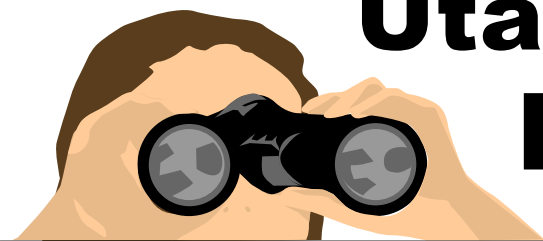
As search engines become better at smelling the rats, websites need to be more informative and rich in content. No longer can a site become a copy of your company brochure.

You can check to see how ethical your own website is by using our free Online Web Optimizer tool at <http://optimiser.123promotion.co.uk>

James Welch is currently MD of 123promotion.co.uk and has worked in the field of Ethical SEO for 5 years after many years as a Software Consultant. He can be reached at jwelch@123promotion.co.uk.

SITE PROMOTION

Terri Wells



Utah Bans Spyware; Fed to Follow?

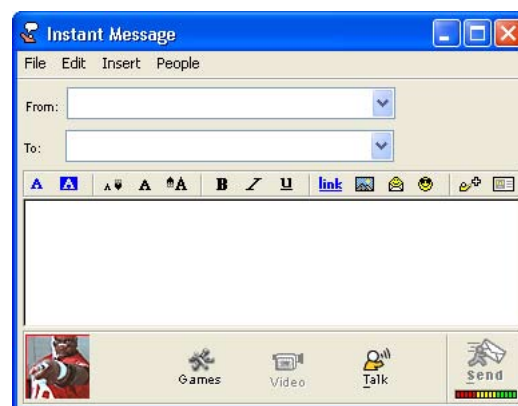
How often does the state of Utah get to make history? This week, during its 2004 general session, it passed a first-of-its-kind bill that made Iowa, California, and even the federal government sit up and take notice. Its House Bill 323, called the “Spyware Control Act,” outlaws certain kinds of software. The definition of “spyware” in the act runs to about 40 lines, and includes (but is probably not limited to) any software that collects information about a user and sends it to a remote location without his or her consent, pops up ads according to the websites a user accesses, obscures content on an Internet website in certain specific ways--in short, the kind of obnoxious stuff that takes the fun out of surfing the Web.

A law means nothing without bite. Okay, so in the state of Utah, a person now “may not: create spyware; cause spyware to be created; install spyware on another person’s computer; or cause spyware to be installed on another person’s computer.” And, incidentally, “It is not a defense to a violation...that a user may remove or hide an advertisement caused by spyware.” So what happens if (more likely, when) someone breaks the law? Then the attorney general may obtain an injunction to prevent further offenses, and recover either actual damages or \$10,000 for each separate violation -- three times that if the offender “willfully or knowingly violated” the law. That’s a nasty little fine, and the good news is, it’s not just the attorney general who can bring an action. The law specifically mentions that injured Internet website owners or registrants, or trademark or copyright owners, may also bring an action. In fact, this law was reportedly inspired when a Utah resident with a business website got complaints about pop-up ads when customers viewed his site. Turns out the pop-ups weren’t his fault; his customers had adware on their computers that reacted to his site, so he took the problem to the legislature.

Before you start cheering, remember that this isn’t quite the black-and-white issue it appears to be when you get those annoying pop-ups every time you check your Web-based e-mail. According to Emily Hackett, executive director of a trade organization called the Internet Alliance, “What’s called spyware is not innately a bad thing. What’s called a pop-up is not innately a bad thing.” She’s right, too. For instance, I know of a ridiculously addictive Web site that lets visitors play its games through pop-ups--click on the game, and it pops right up. Once you’re completely hooked, you can buy the game. That’s

not spyware or adware, that’s good business.

That’s clearly a voluntary use of the software, but what about some of the stuff we’ve been hearing about AOL’s AIM 5.5? The instant messenger has really grown up; now you can play games with it, and even invite your buddy to join you. Well, the game software wasn’t created in-house; it comes from a company called WildTangent, and according to some users, this software shows up as “spyware” when scanned by the right tools. The WildTangent software reports back every time it’s used, supposedly to let the company know what’s popular and what isn’t. The company claims it isn’t collecting personally identifiable information. Is this spyware or good business?



To my way of thinking, that depends a whole lot on whether the users knew what they were getting into. For several years now, “informed consent” when it comes to collection of information -- especially via the Internet -- has been shaping up as a major issue that simply will not go away. More companies “get it” now than did back in, oh, 1999, but obviously more work needs to be done. And for all the screaming we may do about the ones

that are “clueless,” there’s plenty of gray area muddying things up for the honest businessperson who really does want to give his or her customers a better online experience.

Let’s take Microsoft’s dead (and unlamented) Paper Clip in Word as a hypothetical example. This silly animated figure would sit in the lower right hand corner whenever you worked on a Word document and attempt to chime in with helpful suggestions. “It looks like you’re writing a letter! Would you like some help?” You could also ask it for help, assuming it didn’t think to say something first. Annoying and obnoxious—but not an intrusive privacy threat, right? Well, hypothetically, what if this Paper Clip, watching our every move, reported back to Microsoft on what users did—the kinds of help questions they asked, whether they ever took the Paper Clip’s advice, that sort of thing. And yet, what if this information was being used to improve the product—to make it easier to use, say, or offer features that users really want?



Speaking of features that users really want, by some definitions, cookies count as spyware. After all, they’re bits of software that get installed on your computer, track your movements, and report them remotely. Consider this: when companies first started using cookies on the Internet, they made people squeamish. Surfing the Web is such a solitary, anonymous activity that tracking someone’s movements online, even within a single web site, seemed antithetical to the experience. Now, that sounds almost paranoid, especially to folks who’ve visited eBay or Amazon, which couldn’t conduct their business without cookies (even given Amazon’s occasional missteps in that regard).

I mentioned at the beginning that Iowa, California, and the federal government sat up and took notice when Utah enacted the Spyware Control Act. The truth is, all three are considering similar bills. For those who want to check out the federal version, it’s being sponsored by Senators Barbara Boxer (D-California), Conrad Burns (R-Montana), and Ron Wyden (R-Oregon) and it’s referred to as the “Spy-block Act,” short for “Software Principles Yielding Better Levels of Consumer Knowledge.” (One just has to love those acronyms.) I hate pop-up ads as much as the next person -- but I’m all for companies giving their customers what they really want, too. Somewhere in there is, hopefully, a compromise we can all live with.

Frustrated with the ups and downs of search engine rankings?

Let us help keep you on top.



Professional Search Engine Marketing Services

We offer complete search engine marketing solutions including Natural Search Engine Optimization, XML Feeds, Pay Per Click Management and Contextual Advertising.

Submitaweb site, Inc., “The Industry’s Original Submission Company”, has been offering business owners and webmasters leading search engine marketing tools and services since 1997.

We also specialize in providing tools and co-branded services to web hosting, design & development companies and content management system / application service providers.



submitaweb site.com
The Industry’s Original Submission Company

Call us at 480-949-6830

WEB SERVICES: The Web Production Process



Web design and development practices have evolved towards placing more emphasis on usability, fast loading pages, and good quality, content rich web pages. This is done to ensure that anyone browsing a website enjoys a positive experience while visiting the website, increasing the opportunity for better ROI. This article will outline the steps that should be taken when creating and implementing an effective, efficient and creatively unique product while maintaining the proper steps involved in designing and implementing the best possible product.

One of the most common issues web designers encounter is the necessary balance of graphics to straight HTML/Text/CSS. In theory, everyone would like a website only containing pretty graphics and awesome Flash animation. Although we still use these elements on our web page, the proper balance is often overlooked, thus resulting in poorer than expected results from the site, even though on the outside it might look perfect. Additionally, with the various modem speeds used by Internet surfers around the world, webpages must be as small as possible to decrease loading time and increase user retention.

1. Outline the Purpose of Your Web Site.

The first thing a designer needs to establish is the purpose of the website. What market area(s) are being targeted? What age group and demographic? Is the website intended to generate leads? Sales? Answering these questions will help as a guideline to constraints which need to be considered while designing the website. This step in the process is critical to the overall success of the site and will dictate the approach one should take when designing, optimizing, and implementing the web project at hand.

2. Marketing the Web Site.

Once the purpose and target audience of a site has been decided, thinking about the marketing strategy to be adopted should follow. Pay-per-click, search engine submissions, and optimizing the website for various search engines are some of the available options. With the current popularity of search engines,

and the fact that most internet users go to Google or Yahoo! to find what they are looking for, doing some form of search engine optimization will help in gaining more exposure online and reap the benefits of being ranked well on the various search engines.

3. Keywords.

The placement and selection of keywords can play an integral role in the ranking of a website. Tools like Wordtracker (www.wordtracker.com) help identify keywords that might not have been thought of, the competitiveness of those keywords, and how popular they are. Once you have performed this research, try using one keyword phrase per page (outside the necessary pages: about, contact, etc.). This way, you are keeping a common theme throughout your website while providing the user with relevant content. Identify how many keywords you would like to target and create that same number of webpages; you will in essence be creating a keyword rich website.

4. Creating your Site Map.

Proper navigation is important for usability, functionality and search engine spiders. There is nothing worse than having a broken link or unfriendly navigation on a website. Before designing the various webpages, create the sitemap along with outlining the total number of webpages in the website. Grab some paper and pen and outline your entire website structure and navigation. Show which pages link to each other and once again ensure you are keeping your website theme consistent on all pages you are creating in

your sitemap. You can also use this sitemap as another way for the search engine spiders to index your entire site.

5. Copy, Copy, Copy.

Good content is critical these days, both from a search engine and user experience point of view. Today, people are often searching for relevant information using a search engine like Google or Yahoo!. It is critical to create good, keyword rich content that is relevant to the theme and <title> of a web page. Keywords should appear at least 4-6 times on every page that contains text. Users will come back to a website and have only good things to say if their experience is a positive one and they found what they were looking for.

6. Design.

Using the information gathered from the above tips, design the site, using good HTML, CSS, and creative ideas and graphics. Based on the above research, try to maintain the necessary balance between text and html. Utilize the power of Cascading Style Sheets to its fullest and try and keep your individual web pages as close to 100kb as possible so all search engine spiders can view and index your web pages.

7. Quality Assurance.

Testing is the key; be absolutely sure that both the navigation and link structure work perfectly, all graphics are embedded correctly, proper on-page optimization has been done (if required), and that the site works smoothly. Have either a third party Q&A specialist analyze these aspects of your website, or try getting a friend to have a look through the site for navigational purposes. It is always better to have someone review your site for quality assurance who was not involved in the design process, since they're more likely to spot errors and discrepancies. Remember: websites that have broken links can not only annoy your client, but search engines as well.

8. Implementation.

You are now ready to launch your website. Remember that you don't want to upload your index page until you are ready for the search engines to spider your site. There is always the chance a search engine spider will find your site, and the worst thing you can do is turn an incomplete web page with errors live. Create a temporary directory where you can test and view your pages until they are ready to go live.

Designing a website is more than whipping together 10 pages of graphics and text. Today's websites need to look good, as well as have a purpose. In order for a website to be effective in whatever it is understood and expected to do, it needs proper planning and research. The ultimate goal of any web designer is to create a site which captures the user's attention and also provides relevant information that is useful to the person viewing the site. In the end, following these steps will only make life easier on you when debugging, updating, and maintaining these websites along with producing a top quality product.

Cliff Ritter is president of Logical Sense (<http://www.logicalsense.ca>), a Web site design firm specializing in search engine optimization, link strategies, keyword research, analysis, and implementation, along with graphic and web design services.



Looking for a new
web hosting provider?
try **Jumpline.com**

Toll-Free Customer Service

**Quick and Accurate
Technical Support 24/7/365**

Reliable Servers with 99.9% Uptime

No Long Term Contracts, Pay Monthly

**Cutting-Edge Virtual Dedicated
Server Technology**

Customizable Reseller VDS Solution

**HOST 6
WEBSITES
WITHIN
1 ACCOUNT**

VDS500

**MOST
POPULAR**

**500 MB of Storage
10,000 MB of Transfer
150 Email Accounts
Host 6 Websites**

Starting at **\$1495/mo**

visit <http://www.jumpline.com>
for more information.

or call us @

1-800-651-2028



Jumpline.com
establish your domain

Starting a Business Best Practices to Follow

Romeo Marquez

I started a small web design company when I was still a student of Computer Science, here in Monterrey, Mexico. For the past eight years, I have been fully active in running my own web studio, while in a management position with the web development department of one of the largest media companies in Mexico. During this time I have learned many ways to improve my business practices. Dealing with clients, designers, programmers, etc. requires many skills, and not all of them come naturally. In this article, I will share these and some other pointers on how to successfully build stronger relationships with your clients.

The Internet, Now and Then

It was around 1996 when I started HTML programming; the Internet was far from what it is today.

Even though the city I live in is far more technologically advanced than the rest of the country, I had many difficulties finding companies who were willing to spend money on a website.

At the time, it was hard to sell a website because for some people it was hard to visualize the positive impact of having that feature for their company. Selling tools or services online (to their clients) was totally out of the question!

Today it's somewhat easier to find companies with a genuine need for an excellent website. Some require e-Marketing, an intranet, interactive CDs, online catalogs, e-Learning tools, etc.

As the global economy has changed, massive layoffs have become the soil for an impressive amount of small companies and freelancers willing to do these projects; sometimes at astoundingly low prices, making it so that winning project bids is sometimes pretty tough.

So, in an economy where the cake has gotten bigger, as have the number of invitees to the party, what will make you the winner?

There's a constant need for companies and individuals to learn new technologies and keep up with the changing times – there is no doubt about that. There is also a critical need to learn better techniques in finding clients, selling projects, managing and delivering these projects on time and in the best possible way to clients.

So how can you survive in a market where the projects are being disputed by a handful of developing companies and freelancers? What will make any company with a tight budget in need of an application take a look at you and decide to give you their money? Why would anyone trust you?

You shouldn't need an Armani suit to convince anybody. It helps, but people want to know for sure that you can handle their project; that is their real concern. Uncertainty is in any case your biggest enemy.

Unfortunately we live in a world where people judge you by the way you look. You may be the best programmer, the best

project manager, or the best graphic designer, but if you do not present yourself in an attractive and professional way, your chances will usually be diminished.

Habits That Can Make You Very "Attractive"

When I talk about presenting yourself in an attractive way, I don't necessarily mean the Armani suit. A display of habits, such as organization, good communication skills, being punctual for every meeting, along with your excellent coding or your unique design ability can really jump-start the trust relationship.

These habits, among many others, will definitely give a good impression. They make you look professional and project you as being a reliable person. If you communicate well but can't make it on time to your meetings you are most likely to be rejected. So, the more of these habits you have, a better option you'll become to clients. If you don't possess any of these qualities, you better start working on them! As I said before, it is all about the way you present yourself.

I will recommend a few good habits that have helped me grow as a person and a business man.

Don't Ever Give "No" For an Answer

I honestly dislike people that at any request will think twice before saying "maybe," think a million times before saying "yes," but in the split of a second will say "no" a thousand times.

One thing I learned from the only boss I've ever had is to never give "no" for an answer. There is always a way to do something. It might not be the cheapest, fastest, or the best way to do them, but for anything in life, there is always a way to do something.

When a client asks for anything (ridiculous as it may seem) you should always assume there is a way to do it. You might have to do some research, ask a few buddies, or even go to a forum on line, such as forums.DevArticles.com. In any case, you should always try to find an answer to the matter at hand. If you do end up providing a solution, it is now up to the client to decide whether they take it or not. This makes a big difference, because you didn't say "no" and, whatever the outcome, it's the client's choice.



I have sometimes found clients that request something from a service provider and have received the famous “that can’t be done” speech. The service provider was, in most cases, probably right; however, the client who receives a “no” without having the possibility explored will feel hand tied.

When I provide what I consider possible solutions, the client can dismiss his own idea, or take the risk, but at least now he is aware and can make an informed decision. Being positive is a very desirable quality in any service provider.

Keep Up the Good Communication

There’s nothing worse than not knowing how the project is doing, especially at a time when information comes and goes so easily. I usually prefer phone calls when I communicate with my clients. This not only gives me a chance to talk about the things I need to talk about, but also shows that I’m there. This can definitely help to build a stronger relationship with a client.

Any call should always begin with a simple courtesy: ask if they can take a call at the time. Apply this rule whether you are dialing direct phone number or whether you had been screened previously by a secretary.

An email is very effective at times when a phone call cannot be placed or when the client was not available. I would start an email by saying something along the lines of “I called but could not reach you,” and then proceed to discuss whatever was meant to be discussed over the phone.

I highly recommended taking notes during all your meetings,

and as soon as you get to your office, send them in an email to all the meeting attendees. This way, whatever information you gathered from that meeting will be available to anyone involved. Not only will this allow the information to go through an implicit validation process, it also shows good communications skills, and makes the client feel that you are interested in them and their project.

In Conclusion

I know there are many people right now starting their own web-developing business. It may be by yourself or in association with some one else; it doesn’t really matter which. Nor does it matter whether you want to develop websites, Flash design, or interactive CDs. What matters the most is that you take care of your clients in the best professional manner possible and that you keep adding new ways to accomplish the building of a relationship.

In short, treat your clients with a lot of care. Keep them informed and keep the projects on time and within budget.

Romeo Marquez is the owner of *Gelattina.com*, a web application/design company, and a technology consultant. He’s been developing web applications for over 7 years. He can be reached at romeo@gelattina.com



APRIL CONTEST



This month’s contest is a pretty special one for us--probably because this is the most expensive item we’ve ever given away! This month, we’re giving away the Logitech Cordless Desktop MX (for bluetooth), the same one recently reviewed at Dev Hardware (find the review at <http://www.devhardware.com/c/a/Input-Devices/Logitech-Cordless-Desktop-MX-for-Bluetooth/>). Wanna know how to get this awesome prize? Here are the details.

1) Write and submit an article to norbert@developershed.com, with the subject “Developer Shed April Writing Contest” for publication in any of our websites and win this awesome device. This should be a tutorial covering one of the following topics: ASP, Apache, Cascading Style Sheets, COM/COM+, .NET Infrastructure, Google, HTML, IIS, MS SQL, MySQL, Perl, PHP, Search Engine Submission, VB.NET, Web Site Promotion, XML. The article must be at least 1500 words in length, subdivided into at least 5 pages of information. (Check out this article for an example of what we’re looking for.)

2) The Contest runs until April 30, 2004, so your article must be in by that time. Articles may be submitted in the following formats: DOC, RTF, TXT, and HTML (please do not use the “Save as HTML” function in MS Word or OpenOffice). If there are images in the story, please send them in a .ZIP file containing both the pictures and the article. Note that images cannot exceed 400px in width. Articles (and inquiries) should be sent to norbert@developershed.com, with the subject “Developer Shed April Writing Contest.”

3) Judging will be based on subject matter, quality of content (so make it interesting), and grammar. In fact, chances are that if we have to edit it, you ain’t winning. (Ok, so that’s a bit of an exaggeration.) It might still be published, so not all is lost! You may yet still have the chance at the fame and geek chic that comes with having your writings read by millions worldwide. But I don’t want to have to spend three days correcting grammatical mistakes; I’m not your English teacher, daing it!

By submitting an article you agree to give Developer Shed exclusive publication rights to the story. All the credit for genius, however, will stay with you, so no worries about my placing our grubby little name on the story and claiming that it’s our brainchild. The winner will be announced on the website and via phone call, e-mail, IM, carrier pigeon, or smoke signal--which ever is easiest to reach him/her at.

Start your typing!!!

grep this *



Red Planet: Meet Red Planetoid

Sedna, named after an Inuit sea goddess, is a planetoid recently discovered floating just past Pluto, in the Kuiper belt. It's not known why the planetoid is red, but due to its barely-smaller size, Sedna puts into jeopardy Pluto's "planet" status. Sedna is now as close to Earth as it gets, at 8 billion miles, so if you squint really hard, you still won't be able to see it.

Apex on the Move



Apex, known for its low-cost electronics, is making waves in the computer entertainment industry. The company has announced that it plans to release an entertainment PC (US\$799) later this year. The PC will be designed to work with televisions, and will play and record DVDs, serve up digital music, record TV programs--all while being controlled by a remote control, not a keyboard or mouse. Before it is released, Apex will debut its DVD/computer gaming system, ApeX-treme (because no gaming system can be cool without a big "X" in its name).

Dialing for Dates

Say goodbye inefficient flirting! Or so hopes Serendipity's inventor, Nathan Eagle, a computer scientist at MIT. "It's really for people who aren't so great at chat routines," he said, describing his frustration at the inefficiency of chance meetings. It works by having your cell phone to send out (silent) mating calls to others who sign up for the service. No word yet on whether Serendipity will include a "gay-dar" option.



Do Video Games Dream of First Shooter Sheep?



Maybe, if UPN's new series "Game Over" has anything to say about it. The show is about what happens to video game characters after you're done forcing them to die. Characters include a racecar driver, a gun-shooting archeologist (no, not that one), first-shooting ninja neighbors, and drunken, womanizing, Pac-Man-like pets. Sounds like they're living my previous life in *The Sims*.

Spam Lovers of the World, Unite!



"I'm a Spam lover...It's always on my mind... It tastes good, even when cold!" These were the words of Philip Abadilla, founder of the world's first Spam restaurant in Manila, the Philippines. Spamburgers, spam nuggets, green eggs and spam--it's all here. It turns out that Filipinos eat 1.25kg of the stuff annually, much to the chagrin of Monty Python fans. Don't laugh: Soviet Premier Nikita Khrushchev was once quoted as saying that "without spam, we wouldn't have been able to feed our army."

Let's Take This Outside...

Technovoyeuristic Brits have a new hobby: Dogging. The pastime--combining sex, exhibitionism, hot-rods, crowds, public places, and technology--attracts folks of all ages, mostly middle-class, middle-aged heterosexuals. Organizing meetings over the 'net, doggers exchange pictures, and later meet in person to put on a voyeuristic show. They're joined by crowds of onlookers, shoot video, and take photos for the Net. Ten-thousand years of technological evolution and we get...this? No need to pack pants, people!





ADVERTISER index

BSA:

Report Unlicensed Software. [P.9](#)

Elance:

Savvy developers use Elance to profitably outsource projects. [P.13](#)

Developer Shed Weekly Newsletter:

Sign Up Today to stay up-to-date. [P.18](#)

Global Servers:

VPS hosting starting at \$14.95/month plus free domain. [P.12](#)

Jumpline: Looking for a new webmaster provider? [P.23](#)

.Netcharging.com:

Free Developer version. [P.10](#)

Plug In:

Advertising in Plug In just makes goood sense. [P.16](#)

SubmitaWebsite.com:

Frustrated with the ups and downs of search engine rankings?

[P.22](#)

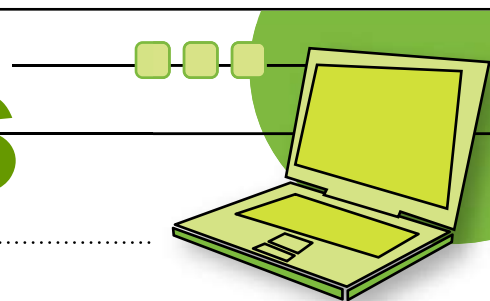
TechSmith:

CAMTASIA STUDIO™ If easy screen recording is where you begin, imagine where you can go. [P.2](#)

TrainingHott:

Web Developer Certificate. [P.14](#)

DEVELOPERS resources



Close Personal Friends of Developer Shed

Behold the power of PHP.

<http://www.zend.com/developers.php>
Zend

Decorating your bookshelf with animals

<http://www.oreillynet.com/>
O'Reilly Books

Required reading for the Information Age

<http://www.osborne.com>
McGraw-Hill/Orborne

Development guides from industry experts

<http://www.oreillynet.com/>
O'Reilly Network

MSTech

Microsoft .NET Framework Development Center

<http://msdn.microsoft.com/netframework/technologyinfo/default.aspx>
Microsoft

Java API Specifications for J2EE

<http://java.sun.com/j2ee/1.4/docs/api/>
Sun Microsystems

.NET Security

http://www.gotdotnet.com/team/clr/about_security.aspx
GotDotNet

Windows XP Security Recommendation Guides

<http://nsa2.www.conxion.com/winxp/>
National Security Agency, USA

Open Source

The most GNU/Linux on 1 floppy disk

<http://www.toms.net/rb/>
tomsrtbt Home Page

Various Open Source Databases

<http://www.mysql.com/>
<http://www.sapdb.org/>
<http://www.postgresql.org/>

Database Comparisons

<http://www.geocities.com/mailsoftware42/db/>
<http://www-306.ibm.com/software/data/pubs/pdfs/bloor.pdf>

Hardware

AMD64 Whitepaper - http://www.amd.com/us-en/assets/content_type/white_papers_and_tech_docs/30172C.pdf

Intel Xeon with 64-bit extensions Whitepaper - ftp://download.intel.com/technology/64bitextensions/IDC_Intel_Xeon_Whitepaper.pdf

Site Promotion

Is your site ethically promoted?

<http://optimiser.123promotion.co.uk/123promotion>

Learn to promote your site the right way

<http://www.seoachat.com>
SEO Chat

Web Services

W3C's Web Services Activity Page

<http://www.w3.org/2002/ws/W3C>

Google Web APIs

<http://www.google.com/apis/>
Google

Numerous articles on Web Services

<http://www.devarticles.com/c/b/Web-Services/>
Dev Articles

shutdown -h now

Katharine Miller

TECHNOSEXUALITY

Hardcore computer enthusiasts have heard all the usual terms of endearment and derision that go along with “computer guy” status: geek, nerd, dweeb, technophile, computer enthusiast, gadgeteer, techie... oh, the list goes on. But just when it seemed safe to wave those geek flags, and march through the streets chanting “we’re geeks, we’re l33t, get used to it,” Ricky Montalvo has popped onto the scene to plant a freshly polished name in the minds of geek fans everywhere: **technosexual**.

The official definition for technosexual is “a dandyish narcissist in love with not only himself, but also his urban lifestyle and gadgets; a straight man who is in touch with his feminine side but has fondness for electronics such as cell phones, PDAs, computers, software, and the web... With metrosexuality, it’s about style, fashion, culture, and grooming for the straight male. A metrosexual man may be seen at an NBA game one night and an art gallery opening the next,” says Technosexual.org’s Ricky Montalvo. “We take it one step further by adding technology. A technosexual man may not need to go to the NBA game because he can get highlights and scores...by browsing the web on his PDA while at the art gallery.”

Despite the *frou frou* connotations, Montalvo insists that this term is not bound by sexual orientation. “The technosexual lifestyle is universal, it has no druthers towards sexual preference... Our guys dress from Banana Republic, Saks, and Neiman’s, shop at Crate & Barrel, but they also make stops [at] Best Buy and CompUSA.”

According to the Alt.Sex.Fetish.Robots FAQ (asfrmcfaq.frostelf.com), “Technosexuality is a word used to get away from the oft-confusing acronym, A.S.F.R., which refers to the sexual interest in machines, robots, androids, gynoids and other sexual devices not really occurring in real life.” Montalvo aims to separate the word from its original meaning and integrate it into the mainstream. “I would like to redefine the term as to a lifestyle where gadgets are a part of everyday style and life.”

This concept hinges on the notion that “lifestyle” encompasses stylish décor, dress, and a high-class standard of living. It also assumes that today’s geek has the income to afford Pottery Barn furnishings, gourmet foodstuffs, and the Sony Clie. Real geeks, however, may forego the furniture in favor of Hitachi’s 300 GB hard drive and a couple of cases of Jolt Cola.

Enter Montalvo’s pet project, Technosexual.org. The website, for now, is strictly an online guide for guys who are technosexual or want to become more in touch with their technosexuality. Montalvo explains the mission of the site, “For geeks in need of style, we are here to say there is hope without losing your geek. We are here to help you polish your lifestyle by adding just a bit more style and flair.”

With sections on style, culture, home, and etiquette, Technosexual.org appears to be a geeky Internet version of the

Queer Eye phenom. “Technosexual can help in any area really,” Montalvo continues, “For cooking, we can recommend cool digital devices to help you prepare a meal. Or for home décor, we can recommend a software program that will help you pick out which colors to choose for the living room.”



Is technosexuality a passing trend or is it here to stay? Montalvo seems to believe in its longevity. “Gadgets have tipped into the mainstream. I just don’t see a world without technology, specifically devices to make our lives easier.”

Going one step further, he suggests that technology should change a woman’s standards when looking to pair up with a member of the opposite gender. “Back in the old days, the woman’s ideal man would be described as someone who [could] mow the lawn, change the oil in the household cars, and clean the rain gutters. Nowadays, a woman’s ideal man should be someone who can trouble-shoot the household DSL or get last-minute movie show times on his PDA while on a date,” Montalvo says.

What about the female technosexuals? Montalvo admits, “Women have always had style, so I tend to think that they would define their own movement, which I would love to see. If [they] were working wirelessly at a café, pulled out a PDA, or mentioned the word Bluetooth, then maybe I’d say [they were] headed in the right direction.”

Until the great FemGeek revolution, feminists seeking out geek-equality can either insist on redefining technosexual once again, or adopting the suggested “technosexualista” or “technodiva.”

Hardcore technophiles will likely disregard the technosexual label. It’s unlikely that production will cease on geek paraphernalia or that Think Geek (www.thinkgeek.com) will switch to Think Technosexual anytime in the future. Technosexual is a fun word and will most likely be thrown about by the same people who discuss hipsters and metrosexuals, whether in satire or in sincerity. But mostly I suspect geeks will use it to meet girls.

Katharine Miller
is a freelance writer in Orlando, FL.
To read more articles or view her silly robots, visit
<http://www.blissfullybitter.com>

