

Frank Hayes

iPhone? Why Not?



IT'S STILL not good enough. That's the reaction of IT analysts and security outfits to Apple's new iPhone 3G. Sure, the iPhone 2.0 software will support Microsoft Exchange and Cisco VPNs. But is it safe enough for enterprise use — as safe as, say, PCs? Gartner says not quite. The security guys say be afraid. It's just not good enough yet.

And it never will be. Oops, that wasn't supposed to slip out.

But hasn't that historically been IT's official position? We're the Department of No. Whatever it is, we're against it.

Cell phones? Wi-Fi? BlackBerries? Web sites? LANs? Laptops? Spreadsheets? PCs? Departmental minis? Not one of those technologies was secure enough, reliable enough and enterprise-ready enough when business users first insisted on sneaking them in under the IT (or MIS or DP) department's radar.

Of course, users had to sneak that stuff in. They knew what the answer would be if they asked us: No. Not ready. Not good enough. Not yet.

And if it was up to us, not ever.

Somehow, though, in the end those users got their way. We made our peace with those unfamiliar, much-dreaded technologies. We figured out how to lock them

down, cordon them off or keep them under control.

Thus, today we hear analysts and security vendors advising us to avoid the much-dreaded iPhone because it may not be as secure as the (once much-dreaded, now familiar) PC.

Ironic? Sure, but we all know the fear that drives that irony. Security and reliability *are* important, and we already have enough trouble on both those fronts. PCs are the devil we know. We're stuck with them. And we can still nurse the fantasy that we can keep iPhones out by just saying no.

But why? Why turn users into outlaws again? It

■ **Why turn users into outlaws again? It never works. It just reinforces the idea that we're the Department of No.**

never works. It just reinforces the idea that we're the Department of No.

Besides, aren't we a lot better off demonstrating that we're the Department of Know-how?

When users come to us with the new iPhone — and now that its price has dropped from the stratosphere to \$199, a lot of users will be coming to us — we don't have to tell them they can't connect to our systems.

Instead, we can start out by asking them what they want to do.

They might ask for the moon. Or they might have more modest ambitions, nothing that would test our systems' security or reliability at all.

Once we know exactly what they want, we can evaluate what will be required and lay that out for the users.

Some of the requirements will be technical. That's our job, and we'd

better be up to it. Another part may be budgetary — someone always has to pay the bill.

Yet another requirement is patience and involvement on the part of those users. To ensure that the result is reliable and secure, we'll have to take a one-step-at-a-time approach. That means users will have to tell us what they want first, and what's less important. They'll have to cut their wish lists and do some waiting. They'll have to prioritize, make decisions and accept compromises — just like we do with every IT project.

Put simply, they'll have to become active partners in the iPhone implementation project.

That won't be a completely satisfactory process for users champing at the bit to get their new iPhones running all over our systems. It won't be quick enough, or let them experiment as much as they want.

But it'll sure beat a blanket denial from the Department of No.

And what do you bet that — like the iPhone — it'll turn out to be good enough? ■

Frank Hayes is Computerworld's senior news columnist. Contact him at frank_hayes@computerworld.com.

Frank Hayes

Reasons to Go



training sounds a lot more cost-effective.

Career dead end. You know all those people you can't promote out of their current jobs because you'll never be able to replace their skills — and can't pay what they're worth because they've topped out the salary range for their jobs? Put them in charge of finding and training their own replacements.

No life. Insane hours, death-march projects and exhausting demands are just the way IT is, right? Baloney. They're just signs of badly managed IT operations and projects, and they're costing you dearly. So stop the madness.

There are better ways to do almost everything in your IT shop — and the people to look to first for ideas are your employees.

Would flextime make them more effective? Telecommuting? Better technology? Improved training? Things you'd never think of? Ask. Learn. Figure out what's possible, then ask some more.

Get rid of the reasons your staff wants to leave and you won't just have a better place to work in IT.

You'll have a better IT shop. ■

Frank Hayes is Computerworld's senior news columnist. Contact him at frank_hayes@computerworld.com.

WHY WOULD ANYONE ever want to leave? As you read about the 100 Best Places to Work in IT in this issue of *Computerworld*, that's a question worth keeping in mind.

No, not about those best places — they're stuffed full of reasons why employees stay. But what about your shop? Why would your employees want to jump ship?

Look, turnover is expensive. The real cost of replacing an employee can run as high as 150% of a year's salary once you add up the costs of recruitment, training and waiting for the new hire to get up to speed, plus the lost knowledge, damaged morale and extra work that comes when any employee bails out. That's on top of payroll — it's like paying for 1.5 phantom employees for every one that leaves.

Average annual turnover for those 100 Best Places to Work? It's a mere 7% — half the industry average. Being a great place to work turns out to be like money in the bank.

OK: What are the reasons your IT people would want to leave? And what can you do about them?

Low pay. There may not be a lot you can do about salaries and raises; they're probably watched like a hawk by your chief finan-

cial officer. But you have options. Example: Stop using your bonus pool as a way of supplementing salaries across the board. Instead, pay much bigger bonuses to fewer people for specific, publicly recognized accomplishments. Suddenly, a bonus means something. Everybody on your staff won't hit the jackpot, but everyone will have a chance — based on what he accomplishes.

Boredom. Sure, our job isn't to entertain our IT employees. But if someone is actually bored, something's wrong. Maybe it's the wrong job for him. Maybe it's the wrong company. Or maybe he could

■ **Turnover for the 100 Best Places is half the industry average. Being a great place to work turns out to be like money in the bank.**

be doing a lot more, but the current job won't let him. Hey, if he can polish off his week's regular work in 30 hours, reward that with opportunities for new projects, training, coaching fellow workers — whatever will motivate and challenge him.

No training. Yeah, we all know the excuse: If we pay for training, employees will just take that new knowledge and leave for better jobs. But with no training, they'll get fed up and leave anyway — or rot away with outdated skills. So get creative. Pay for training with loans you forgive only after a certain number of months or years. If an employee is willing to pay out of his own pocket but needs time off for training, be as flexible as you possibly can. And compare the cost of retraining a current employee with the real cost of a new hire. Suddenly,