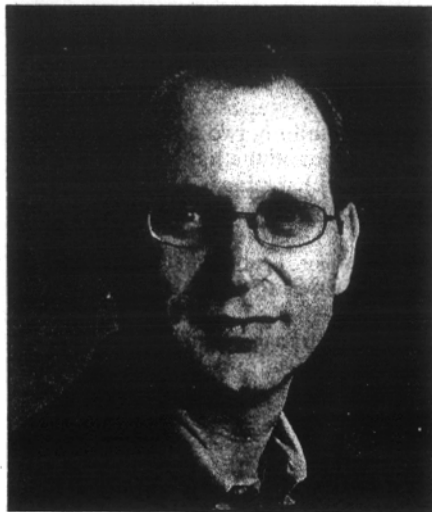


# ROCHESTER BUSINESS JOURNAL

## Author Glen to speak about leading the geeks



Paul Glen

By ALEXANDER SOULE

The dot-com boom has come and gone, but the geeks still keep the technology running at companies, and need to be managed.

Author Paul Glen is slated May 2 to speak to the eBusiness Association on "Leading Geeks—How to Manage and Lead People Who Deliver Technology" and at the University of Rochester.

Rochester Business Journal reporter Alexander Soule interviewed Glen last week about building effective technology organizations.

Glen is a management consultant, author and speaker who helps clients build effective technology organizations. He wrote two books: "Leading Geeks: How to Manage and Lead People Who Deliver Technology" and "Healing Client Relationships: A Professional's Guide to Manag-

ing Client Conflict" He also writes a monthly management column for the online edition of Computerworld.

An edited transcript of the interview follows.

**ROCHESTER BUSINESS JOURNAL:** How has the downturn in technology spending affected how managers deal with their technical work force?

**PAUL GLEN:** In general, how managers deal with their technology work force is not radically different today than it was before the dot-com boom. What has changed, now that the dot-com boom and bust have come and gone, is that some of the silly ideas about how you deal with ordinary employees are now different.

The idea that stock options are the only way to reward employees is gone. The popular image of people getting around the office on scooters and a foosball game

in every corner is no longer with us.

I always considered it a bubble mentality. When I was teaching graduate school, all my students were telling me: "No you just don't get it—you don't understand what it is really like out there." But what I kept reminding them is that 96 percent of technology workers don't actually work in high-tech companies. They work in real companies and deal with real issues like keeping the systems running and getting the bills out.

**RBJ:** What mistakes do CEOs make in dealing with their tech employees?

**GLEN:** CEOs are becoming more and more aware of the strategic role that technology plays in their organizations, but that does not necessarily translate into doing a better job of recognizing the unique people who create it for them.

Continued on page 12



**C2 CONSULTING**  
3253 MALCOLM AVE.  
LOS ANGELES, CA 90034

## GLEN

Continued from page 6

It's fair to say that there are patterns I see repeatedly. The biggest one is failing to recognize that geeks are more loyal to their technologies than they are to their managers. They tend to be more interested in how things work. There are a lot of managers who believe that people are most attached to a business or a personality, and with technology employees it doesn't work that way.

Technologists have a passion for reason. They are drawn to rationality, clarity. Where an emotional pitch might work great for motivating a sales force, it isn't going to work for the technology crowd. You have to motivate, but beyond all else you have to hew to reason.

Tech workers also have a tendency to be judgmental. You are either a genius or an idiot—they don't tease apart the more subtle differences in people. They see everything in binary code.

### Prevailing tech attitudes

**RBJ:** What are managerial attitudes toward technology projects today?

**GLEN:** C-level managers often lack patience to deal with the failure rate of technology projects. Only between 25 and 28 percent of technology projects are completed as planned. The rest are either late, overbudget or don't get completed at all.

That leads managers to lose lots of sleep, and companies to lose lots of money. Managers sometimes have a false comfort level that project management can save them, and they forget about the human aspect.

**RBJ:** What does that do for the psychology of rank-and-file technology workers, when 75 percent of their projects are leaving them open to criticism?

**GLEN:** I would say first that all of your work is opening you up to criticism, but managers have to create an environment

where it is safe to fail. Otherwise people will never try to go above and beyond what is expected of them. If people are like scared bunnies hiding underneath their desks, they won't try to create brave new products.

**RBJ:** What company comes to mind as a model for managing its technology work force?

**GLEN:** I don't know if I could pick out any one company where everything with regard to its work force is magical. More often, within individual companies there are pockets or niche groups where things really click.

I suppose one might say Microsoft. Love them or hate them, they turn out a tremendous amount of code from what is relatively a small staff.

**RBJ:** Is there a way to institutionalize the best practices that you see?

**GLEN:** In general, the militaristic, hierarchical model is a great one for predictability, but it is terrible for creativity. There are a lot of companies out there who think they will get the most out of people by putting them in that kind of an environment.

But hierarchical models don't work because of what I call the knowledge inversion. People always come to the workplace thinking that their boss should know more about their job than they do—it may go back to the medieval guild system, where you were an apprentice before you moved up to become a master and knew everything.

But generally, when you are supervising

a technologist, they are going to know more about their job than you do.

Failure to recognize the knowledge inversion leads geeks to think their bosses are idiots, and managers feel compelled to make decisions on things they don't know much about. That's in part why the hierarchical model is not effective.

### Dot-com lessons

**RBJ:** During the technology boom of the late 1990s, a lot of younger workers were thrust quickly into advanced managerial positions. Are today's companies benefiting from that generation's trial by fire?

**GLEN:** I would love to be able to say that some of those younger executives learned a little bit of humility, but I am not sure I've seen the kind of learning I would like to see out of that. More often I come across a lot of people who simply feel victimized by the whole experience.

If you are a C-level executive thinking about hiring a person

who went through the dot-com experience, it behooves you to try to tease apart what that really meant for that person.

There were people who learned humility, but there were a lot of people who learned hubris. I have seen plenty of people join the corporate ranks who repeat the dot-com mistakes.

**RBJ:** What did we learn during the boom

Continued on next page



Continued from previous page

about what motivates technology workers?

**GLEN:** There is a real misunderstanding out there about what motivates technology workers. By and large, the motivation to be creative does not come from a company's compensation structure. Motivation comes from engagement with work, not engagement with a compensation scheme.

Put another way, if someone gave me 10 million stock options and I spend all my time trying to figure out which Caribbean island to retire on, I am not thinking about code.

Executives can't make someone interested in something. I think of it as nurturing motivation, as opposed to motivating people. An analogy would be gardening. What can you do to make a seed grow? The answer is nothing. You can till the soil, you can water the soil, you can fertilize the soil—that's all you can do.

When assigning teams, many managers aren't asking the right questions. They ask instead who is available, or who has the skills, or who has done it before. They don't ask who wants to do the project. If you want a group of people to be motivated to do a project, select people who are motivated to do it.

Another mistake managers make is that they forget to manage the meaning of the project. We get very wrapped up in facts, technical details, milestones and deliverables, which is not the same thing as discovering the meaning of the project.

A good example I saw recently was UNICEF sending people going all over the world to set up network nodes for connectivity. These people were literally volunteering to go into war zones with flak jackets to do it. For them, the meaning of the work was saving children all over the world.

Too many projects are approached as a process, as opposed to a product. There is something motivational about beginnings and endings. People get excited about making things, as opposed to being a cog in a wheel.

asoule@rbj.net / 585-548-8303