GUEST EDITORIAL

FROM PETER DE JAGER

Dear NaSPA Member;

From Technical to Business Support

The Year 2000 problem is not a technical issue, it's a management problem. You've heard that before, but have the implications set in? Has it changed how you think about your job? Do you now think of yourself as a "Business Support" person, or are you still "merely" a techie?

Without doubt, the Y2K problem is our biggest failure. Business leaders who look at this problem just shake their heads in disbelief. "How could we be this stupid?" is the question they ask over and over again.

We could, and do, try to rationalize how we got to where we are today. The real answer is simple and unpleasant: We got here by not really taking responsibility for technological decisions in our organizations. We were hired as the technology experts, but allowed non-technical people to make technical decisions and nodded our heads in submission as they handed out unrealistic deadlines and constraints.

We lack the courage to say "no"... and then stick to that response regardless of the consequences. But explaining how we got here is not where we should be expending our energy. Our focus should be 100 percent on the small matter of fixing this problem while we still have the time.

Here's the bottom line: The survival of your company now rests upon the shoulders of technical support. Only technical support understands how much the organization relies upon systems and upon the individual lines of code making up those systems. Only technical support knows this code will fail when we start feeding it a steady diet of '00', and only technical support has the knowledge necessary to fix the code.

For better or worse, rightly or wrongly, the responsibility for getting us out of this mess lies with us. We didn't ask for this responsibility, but we have it, purely because we're the only ones with the knowledge necessary to fix the problem.

So now we're faced with a choice. Do we rise to the challenge and fully accept the responsibility of fixing this? Or do we say "That's not our problem! It's maintenance!"

Well, it is maintenance. Possibly the most mind-numbing maintenance project we'll ever have the misfortune to be a part of, but if it's not done, properly and on time, then businesses will fail.

Rising to the challenge will be difficult. We're faced with a task with a totally immovable deadline that cannot be missed. That's never happened before. Deadlines could always be moved and when they couldn't be moved, they could be ignored. Not this time.

This time, we cannot be late. Our historical record of being late on nearly all projects cannot be repeated on this one. We cannot be late. We cannot be late. We cannot be late. That cannot be repeated enough times on this project. Jan 1, 2000 will arrive on time and we had better be ready.

So, what do we do differently? How do we run a critical project and defy the first law of programming "Whatever can go wrong, will go wrong, and it will go wrong at the worst possible time?"And the second law of programming which states that "everything takes longer than expected?"

What we do differently is shelve our optimism for the duration of this project and plan for things to go wrong. We plan for missed deadlines and create contingency plans to take them into account. We apply our significant intelligence to the problem of being on time and everything else is put to the side. In other words, we accept fully our professional responsibility for getting the task done on time, recognizing there is nobody else who can do it... for better or worse, we support the business.

Sincerely,

P De Jager

Peter de Jager