

Re: Do you have a Knowledge Officer?

Re: Do you have a Knowledge Officer?

Source: <http://coding.derkeiler.com/Archive/Cobol/comp.lang.cobol/2007-10/msg00351.html>

- *From:* Robert <no@xxxxxx>
 - *Date:* Sun, 07 Oct 2007 14:48:12 -0500
-

On Sun, 7 Oct 2007 13:37:02 +1300, "Pete Dashwood" <dashwood@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote:

"Robert" <no@xxxxxx> wrote in message
news:0mhfg3pdinc9n3o9g8ln2lqq9h724op8ai@xxxxxxxxxxx

* It must be an environment/configuration/deployment problem: this may be actually true, but it usually points to a larger stability problem. If you cannot build and deploy reliably then why would you have confidence that the code works?

Because the deployment package has nothing to do with the application code?
Duh!

I can sympathize with that excuse. Too often, our change control infrastructure is error-prone, has hidden dependencies, especially humans remembering to do things. The programmer gets blamed for screw-ups outside the development domain.

So, how do you handle being blamed for something that isn't your fault?

Re: Do you have a Knowledge Officer?

(Most of us experience this at some stage, so dealing with it is a useful skill to acquire. Angry tirades, sulks, and standing on your dignity are not solutions...)

Women face this predicament in domestic life. Some just take it; we call them doormats. Some respond with anger or withdrawal. As you say, those are not solutions (for adults; they work for babies). Women with self-respect move out, then find another partner.

High turnover in a computer shop should be a red flag. So should a high percentage of contractors, especially contractors who have been in the same job for years. Contractors have zero political power; they're not allowed to criticize.

My solution is to FIX the change control system .. by automating everything, removing dependence on manual steps, adding verification checks. I love working on infrastructure automation. In my domestic analogy, it's like employing a household organizer or relationship mediator.

Management is responsible for deployment, not developers. The problem is, management sees it as a techie issue, thinks 'let the techies fight it out among themselves.' That's a real problem.

Management are no more responsible for deployment than they are for developing the system, but you don't see them writing code.

IT management has technical people responsible for infrastructure, including deployment.

Again, management is responsible for keeping ducks pointed in the same direction. Workers are not going to do it themselves unless they're given a structure and incentive.

It is part of the Management responsibility to provide that structure and incentive. I am very happy to set goals , explain why we are going that way, agree the direction and goals with the team, then let them get on with it, providing whatever support they need. They are very capable of pointing ducks in the right direction :-).

Managing programmers has been likened to hearing cats.

Re: Do you have a Knowledge Officer?

* It Worked on my Machine!: programmers use this excuse to downplay a bug. The reality is actually the opposite – it means that you have an intermittent bug which is by far the worst kind of bug to have in your application. You want bugs to fail quickly and consistently – any variant such as "That's Weird", "That didn't happen yesterday", "That must be a data problem", etc. is admitting you have a bug that cannot be easily duplicated.

Not necessarily at all. It may well have worked on a local machine and there are simply configuration problems (permissions, authentication, logons, user profiles, dozens of factors that have nothing to do with a bug in the application code.) with getting it networked. It isn't necessarily an excuse for a bug, it is simply a statement of fact. You can't expect programmers to be network administrators and configurators, if you are employing them to write code. That's why you have separate areas of expertise.

Exactly. But the defect is first assigned to the application team. The burden is on them to figure out whose fault it really is.

Who else could do that, given that Management are generally not technical?

If the failure is due to infrastructure, it should be fixed at that level. Application teams shouldn't have to deal with insufficient disk space, dropped database connections, etc.

Re: Do you have a Knowledge Officer?

Re: Do you have a Knowledge Officer?

Many of these problems are caused by fragile environments, which in turn are caused by Balkanization of responsibility. SOMEone should be responsible for the health of the whole organization.

Ah, that would be the CEO... maybe you should be talking to him/her...

If you're talking about IT then that would be the CIO and you DEFINITELY should be talking to him/her...

I do, sometimes. Other times their attention is directed upward, at their bosses. They can't be bothered with technical stuff.

Someone should fix the root cause of 'incidents' so they don't happen again.

That is what tech teams endeavour to do.

Right. That's what they're supposed to do. Too often they act as gatekeepers rather than enablers.

.