

# THE PROBLEM OF Creeping Commitment

BY GLENN CAMPBELL

How can a 200-pound man lift a 2000-pound bull?

Every bumpkin farmer knows it's easy: You start when the bull is just a calf and can be picked up with little difficulty. Every day, the farmer picks up the animal again. The calf gets bigger as the farmer gets stronger. After a couple of years, the farmer can lift the adult bull, no problem.

What's wrong with this theory? It assumes that because each day's increase is almost trivial, it can always be overcome. However, even trivial changes add up. We know the end point is impossible, so somewhere along the line the system has to break down.

This same delusion is often found in well-meaning people who want to help others: Every day, they take on a little more responsibility, usually for noble reasons. Each new commitment seems minor in itself, so they say "No problem." In the end, however, the cumulative burden is unsupportable and the system has to fail.

On some future day, which can't be precisely predicted, the farmer will not be able to lift the bull. Furthermore, when the system fails, it is likely to do so catastrophically: The farmer may break his back or otherwise injure himself by going beyond his limits.

Think of your neighborhood cat lady, whose heart for strays is so big that she can't turn any feline away. Each new kitty is no problem, she thinks, until she has hundreds of cats, living in deplorable

conditions, and the county has to shut her down. In the end, this misguided charity collapses under its own weight, and no cats are really served.

This phenomenon of incremental entrapment can be called "creeping commitment," and it's the official disease of do-gooders everywhere. Doing good isn't the problem; it's knowing when to stop.

In virtually every helping profession, the needs of the clientele far outstrip the resources available. Any conscientious caseworker, foster parent or juvenile probation officer is bound to get overloaded. The alternatives are often unthinkable: If you don't help, then no one else will, and a child may descend into an unimaginable hell.

But the consequences of overload are also dire. The more overcommitted you are, the more you cheat on self-maintenance and on the quality of your product, and the more vulnerable you become to catastrophic collapse.

If your system fails, then you won't be helping anyone, and you may end up sending your entire client list to that unimaginable hell.

Every system needs a buffer. That's a generous helping of empty space where nothing much is planned and flexibility is retained. There has to be opportunity for preventative maintenance, exploration and continuing education. Even gods need time to clip their toenails, if not paint them.

Every complex system also needs the room to evolve and adapt. When the

world changes, you don't want to be trapped in old technology designed for last year's problems. If you get yourself so committed that you can't periodically withdraw and regroup then your mission is going to suffer.

The answer to creeping commitment is to actively shed commitments at the same rate you are taking them on. If you have the impulse to help, you must also embrace "pushback"—the inclination to deflect responsibilities away.

If you take a new cat into your home, you may have to get rid of another. For anyone with a conscience, the taking-in is easy, but the getting-rid-of is not. That's why you have to look for natural boundaries wherever you find them, then actively defend them against encroachment.

Clear distinctions must be drawn between "my problem" and "not my problem." Ideally, a problem should be solved by the party most closely attached to it. Why should I feed you if you can feed yourself?

Unfortunately, this ideal condition isn't always possible, which why help is needed. Any intervention, however, should be as brief and limited as possible. Responsibilities need to be pushed back at the earliest possible opportunity, even if the client doesn't seem quite ready.

This solution may be imperfect, but so is every alternative.

—G.C