

Fear of intervention - how to make subordinates work as entrepreneurs.

by Willem van den Ende, 21 august 2003

Introduction

Increasing the effectiveness of workers in a software development organization can often be done by introducing some surprisingly simple practices. However, as I found out when I tried to become more effective myself and helping others to do so, manager-subordinate relations and a history of less than pleasant management interventions can make workers so afraid of interventions they become sort of paralysed, and can hardly improve their effectiveness. Using an allegorical story based on various experiences and Diagrams of Effects I will investigate some actions that can be undertaken to help someone in a subordinate position to become empowered again and improve his or her effectiveness. As a reviewer pointed out, there are also actions that can be undertaken to make the manager more effective. I will leave that to another article.

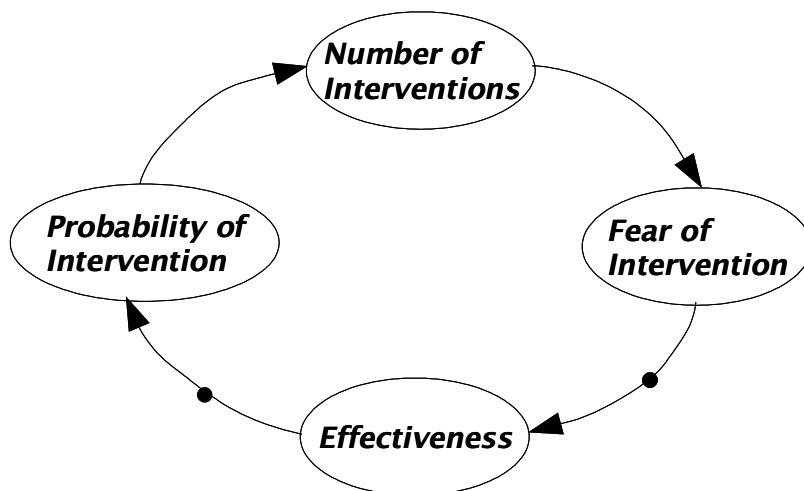
Tom and Rick's history


Tom is a manager who used to apply an authoritarian style of management. He is a senior manager with much more experience than his relatively young employees, such as Rick, who tries to grow from programmer to project leader. Using his experience this way has enabled Tom to make swift decisions so his teams are agile and make rapid progress. The number of Tom's subordinates has been growing rapidly, so he does not have enough time to be involved in all the details anymore. Therefore, Tom wants Rick and his colleagues to behave in a more entrepreneurial fashion; they have to take risk, invent new business ideas, experiment with these new ideas and generally make more decisions on their own.

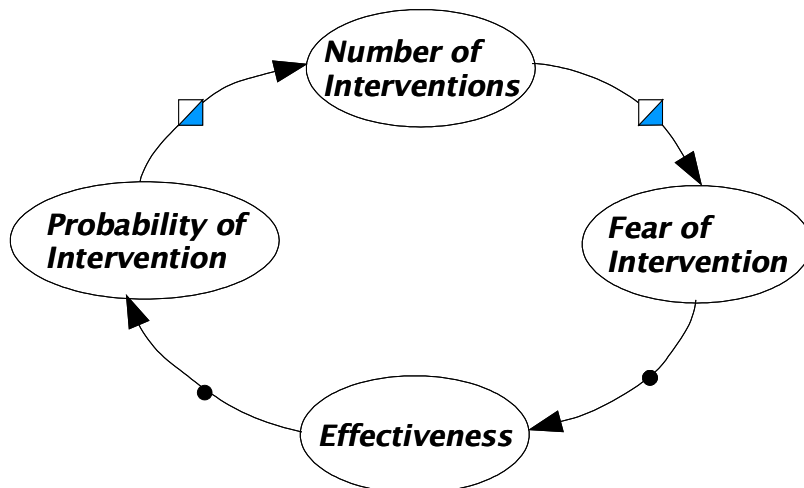
In the past Tom has experimented with giving subordinates more latitude. Sometimes the subordinates failed, so Tom had to intervene. For instance, Rick had hired some programmers who quarreled with Tom a lot. Tom got irritated, so he fired them swiftly without consulting Rick first. Rick was not pleased when he heard most of his development team got the sack. Tom wasn't happy with this intervention either, he just felt he was driven into a corner and had no choice.

With a new process, and training and coaching in several key skills, and more experience the subordinates could be more able to be entrepreneurial, so the manager wishes to give it another try.

Now history becomes a problem. Rick is willing to take on more responsibility, but is afraid Tom might do more unpleasant interventions like before. This fear lowers Rick's effectiveness when carrying out his tasks independently: he procrastinates, speculates about possible interventions, tries to second-guess Tom's wishes. When Rick is ineffective, Tom is more likely to intervene, and with each intervention the fear of interventions with Rick and his colleagues rises. This gives us the vicious circle in the following diagram (The symbol ● denotes a negative influence, the absence of this symbol denotes a positive influence):



So, how can Tom and Rick escape from this spiral? There are two intervention points within the circle. Subordinates can accept an intervention as a signal that they still have something to learn, without letting an intervention result in growing fear. The manager can try to reduce the number of interventions. This results in the following diagram. We use the *management action with open choice effect* symbol  to indicate that someone can choose to let the arrow have a positive influence, a negative influence or no influence at all:



So, both the manager and the subordinates have the choice to not let a history of interventions influence them. Growing this kind of trust takes time and effort from both parties.

Maybe there also is a way to increase the effectiveness of the subordinates, even if we disregard the fear for a moment. The introduction also mentioned training, coaching and experience. Do they have an effect? They probably have, but they do not influence the way people work immediately. Some training is easier applied than others, and some experience only works if one makes the same mistake twice (or even more), reflecting on experience takes time as well.

Before we go onto the next diagram, let's elaborate a bit on possible training, coaching and using the growing experience of Rick and his colleagues.

Possible training includes training on communicative skills, sales, planning. This usually requires Rick to spend some days outside the company in training. After that, Rick needs some time to integrate what he learned into his daily practice. Another option is training on the job by an external coach or by experienced people inside the

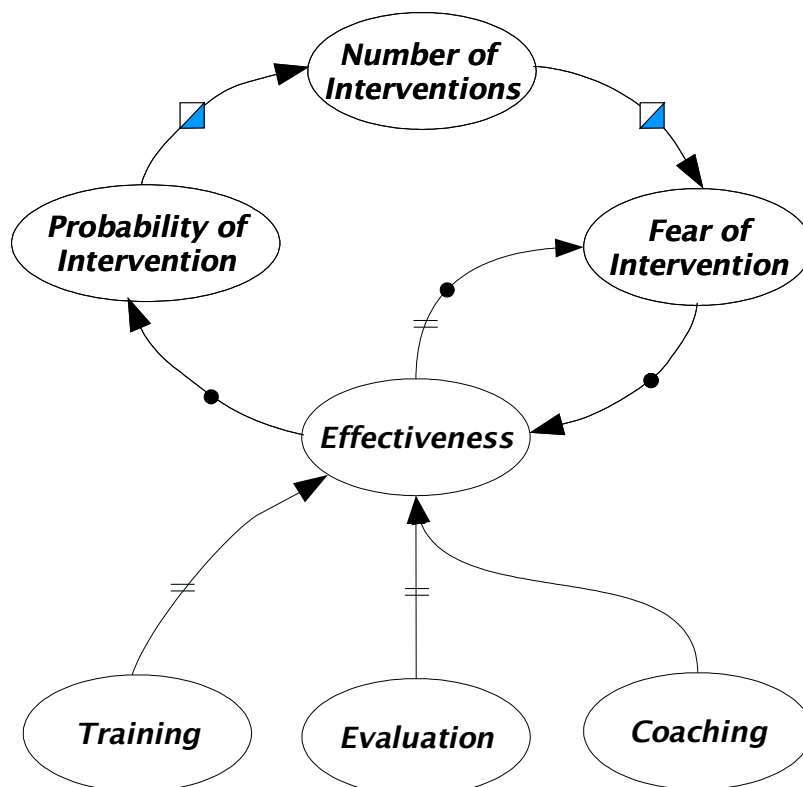
company. This way, Rick can learn while providing value to the company at the same time.

Tom offers Rick the benefits of his expertise. Tom has many years of experience, which he uses to make decisions. Tom can teach some of his experience Rick and others, so they can make better decisions on their own and Tom can delegate responsibility easier. Rick also decides to take some courses on sales, and Tom and Rick decide together to hire Steve, an external coach to help them introduce a more agile software development process in Rick's projects, so Rick and his developers can improve their effectiveness faster.

Working together on concrete projects is a great way to learn, using the experience gained in each project to improve the next project. With their new process, Rick and Tom can run projects of a few months maximum. With Steve's help they run an evaluation after each project, so they and other participants such as customers and programmers can all learn from their experience and increase their effectiveness. Evaluations gradually increase effectiveness, it is not something that happens overnight.

After a few one-month projects Tom and Rick get a better grip on planning inside projects and around projects, and projects start to run smoother, so effectiveness increases. Rick's fear is only slowly decreasing, since the memories of some interventions are still quite painful.

So training, on-the job coaching and regular project evaluations have a positive effect on effectiveness and rising effectiveness will reduce Rick's fear of intervention, which in turn will make him more effective - a virtuous cycle. But, as indicated before, training and evaluation have influence after some time, which we indicate with the delay symbol = .



Training, evaluations and coaching only work, if they combined are stronger than Rick's fear of intervention. For instance, if Steve were to leave now, it is questionable if Rick's effectiveness would improve. What can be done to reduce Rick's fear more directly?

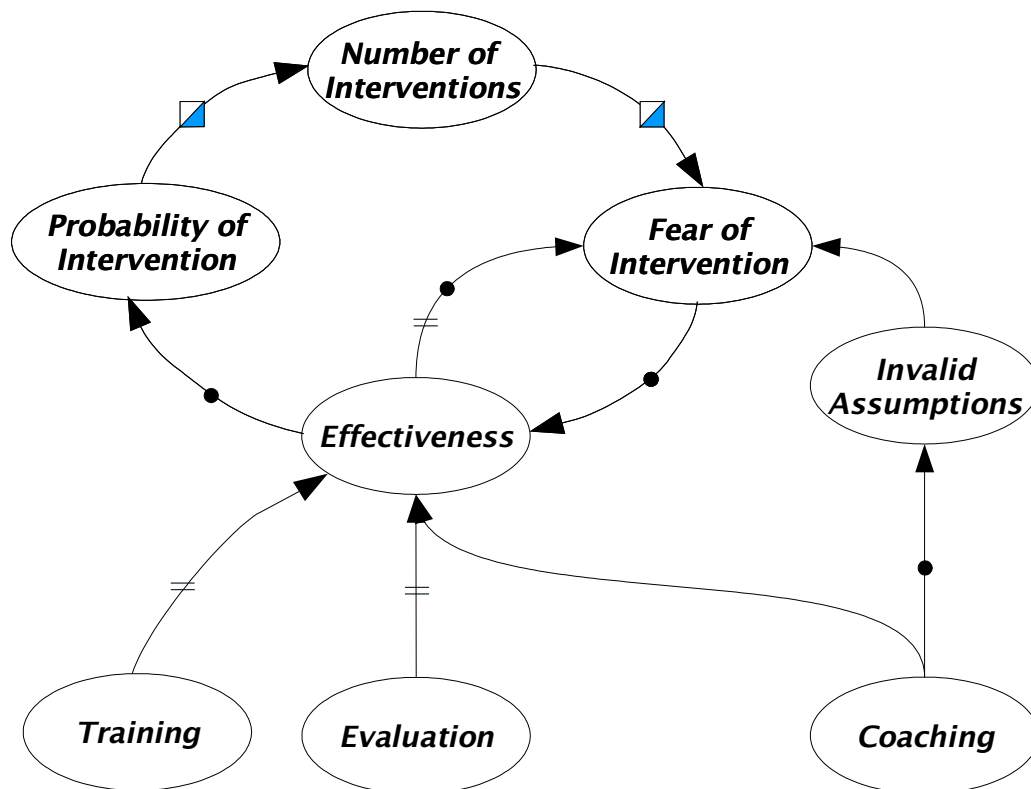
Tom is trying to make the switch from a 'push' model, where he gives orders to

people, to a 'pull' model, where people can ask him for advice when they need it. For Rick and Tom this means they have to keep a delicate balance. Rick can ask questions but has to make his own decisions. Tom has to make a big effort to prevent his suggestions from being perceived as orders. For instance, sometimes Tom will keep his opinion to himself, in order to give Rick space to do his own thing, Rick might erroneously perceive this as disinterest by Tom... One way out of this dilemma is for Tom and Rick to make their assumptions explicit, share their fears, hopes and expectations.

One time, Rick and Tom have a meeting where Steve, the coach, is also present. Steve notices Rick is getting angry at Tom. Tom tries to coach Rick and get him to explore several options for the process to use in their next project. Rick assumes Tom is trying to tell him what to do, and Rick starts to resist Tom, because he wants to do the project differently. Steve decides not to say much, but just ask Rick a few questions to make Rick aware of his assumptions. Even though Tom had a strong opinion on the issues, the final decisions would be up to Rick. After a while Rick's anger dissipates, and he and Tom have a constructive discussion.

After the meeting Rick asks Steve why he didn't say very much and use his weight as an 'process expert' to influence the meeting. Steve says, that like Tom he has a strong opinion on how he likes to run a project, and that he thinks differently about it than both Tom and Rick, but that it is up to Rick to run the project and find out, as much as possible through his own experience, how he likes to run a project. Steve says he is glad to give his opinion and advice, but only if Rick asks for his help. From his experience Steve knows, that if he gives advice to someone who did not ask for help, there is a big chance his advice will be ignored, or even worse, that the person he is trying to help feels insulted or threatened, and does exactly the opposite of what Steve suggests.

Steve explains to Rick that Tom was trying to do the same in the conversation they just had, give advice only if someone asks for help, and not use his weight as an expert and boss to give orders. He advises Rick, to be more aware of his assumptions, and validate them more often, so Rick does not become angry or afraid because he thinks Tom is intervening unpleasantly again, when Tom is only trying to help Rick.



So the diagrams tell us, it is possible for Tom to delegate more responsibility to Rick and his colleagues and for Rick to grow and assume the responsibility and behave more entrepreneurial. To get there, there are some things that need to be done: Tom reduces the amount of interventions, and invests time in coaching Rick and his colleagues. Tom and Rick reserve time and money for training, evaluation and coaching. Rick and his colleagues realize that when they are becoming more effective, there will be less interventions, and welcome an intervention, if any, as an opportunity to learn. Rick learns to validate his assumptions before he grows afraid. Rick also takes time to follow training and coaching, and he makes sure everyone in his projects participates in evaluations. But wait a minute, now Rick is reading this, he realizes something....

Rick and Tom are not that different

Rick used to be a programmer, now he is leading projects. Rick is an expert programmer, so his favorite way of leading a project is looking at the code and seeing what happens there. If he doesn't like some of the code, he makes a suggestion to the programmers to do it differently. This works well, he thinks, because with his experience the team can make quick decisions, make rapid progress and be agile.

Now Rick has a problem: the process improvement he is doing together with Tom, all the training courses, and organizing evaluations take up a lot of his time. Now he wishes to do the same thing Tom does: delegate some of his responsibility to his programmers, let them make more decisions on their own. But every time he tries to let programmers do their own design, either nothing happens, or they are constantly at his desk asking questions. What can he do?

After some time Rick realizes he has the same problem Tom has. His 'design suggestions' are taken as orders by the development team. Rick rarely explains the rationale behind his decisions, because he is so experienced that he uses the patterns he knows intuitively. The programmers think Rick is really clever, and that

they could never do designs Rick's way.

So, what can Rick do? Apply the diagram to his teams and invest time and money in training, coaching and evaluation, and let the programmers gain confidence by letting them be more effective.

Rick has to do the same thing to his programmers Tom and Steve are doing to him: give advice only to programmers that ask for his help, trust them and give them as much space as possible, allow them to make mistakes and learn from those. If a programmer gets angry or frustrated with Rick, Rick tries to coach the programmer, to find out which assumptions the programmer is making.

Conclusion

Using the diagram of effects as a model for some of the observations in Tom and Ricks story enabled us to find intervention points to break the vicious circle Tom and Rick found themselves in. Training, regular evaluations and coaching are actions that can help Rick to become more effective and less afraid of interventions.

I would like to thank Pascal van Cauwenberghe, Marc Evers, Erik Groeneveld, Nynke Fokma and Peter Schrier for developing the various diagrams with me and their encouragement and feedback on the write-up.

Developing these diagrams together unearthed more possible actions than could fit into a small diagram, I hope these diagrams inspire you to find your own. Maybe, like the reviewers and myself, you recognize yourself in Tom and/or Rick. I used this diagram to understand the fear of intervention, when I encountered it again at a new client – I had been there before, but as Erik says: “sometimes one has to make or see the same mistake twice to understand it...”

Willem van den Ende works as an independent Software Development Coach for Living Software B.V. He welcomes your feedback via mail [@willemvandenende.com](mailto:willemvandenende.com)