Step 5: Develop Countermeasures

Overview of Countermeasures

A proper root cause analysis will usually point to the action needed or least point us in a very good direction. For example, one of the issues the Treetop team discovered during their root cause analysis was that new employees often created defects because of lack of training. So, in this case, the Treetop team must develop countermeasures to deal with this particular issue.

And I stress the word **countermeasure** since we use this term instead of "solution". Why is this important? Well, a "solution" implies finality, an end to the process of discovery and correction. Whereas within PDCA, or continuous improvement in general, there's never an ending to the journey.

So when we do problem solving we're implementing countermeasures to make positive changes to an existing root cause. But this new condition isn't perfect yet, and never will be since, this same problem might occur again meaning it wasn’t solved. However, if the original countermeasure was successful this problem will not re-appear for the same root cause since it’s been countered.

Brainstorming

In some case you and your team may need to complete a brainstorming session in order to develop detailed countermeasures. We’d like to take a few moments to explain the proper way to facilitate a brainstorming session since, done poorly, they can actually be counter-productive.

To start with, we suggest using a flip chart. Then the person facilitating the session will write the problem statement at the top of the sheet while also numbering the sheet at the top of the page.

After this the facilitator will ask person on the team to present an idea. One very important point I’d like to make here is that there shouldn't be any discussion at this point. In other words, there is no debating or elaborating of points allowed at the stage. Instead, the facilitator will write exactly what the person says.

You'll notice I didn’t say the facilitator will paraphrase what the person said since **paraphrasing is very dangerous** during a brainstorming session since ideas are often tweaked and then perceived differently than the person offering the idea desired.

Once the team seems to have exhausted their ideas…. it's time to discuss the list in order to better understand what someone meant. After this brief discussion teams will often have additional ideas and the process can continue until there are no more ideas.

And finally, one other tip I’d like to share is to **never make a particular idea standout** from the rest of the ideas by bolding it or underlining it as this can make others feel as if there ideas are not good, essentially shutting them down for the rest of the session.
Selecting Countermeasures

Alright, well once you and your team have a list of ideas it’s time to select the countermeasures to start with.

To do this there are a number of different ways to go about this. For example, we can also use the \( N / 3 \) technique as we learned about in a previous module.

Another powerful tool that can be used to select the countermeasures to start with is called in impact / effort grid. With this tool we ask ourselves how much effort will be required to implement the countermeasure as well as its predicted impact.

Then, as an example, if we predict a countermeasure will be high effort with low impact this might suggest we look at another idea. The best scenario occurs when there is a countermeasure requiring low effort to implement while returning a high impact.

A slight variation to this matrix occurs when we replace effort with cost turning it into a cost/impact matrix. Many organizations use both versions to help them select the best way forward.

Set Based Countermeasures

Now then, while we’ve been discussing how to narrow down your list of countermeasures I’d like to qualify one thing… namely that our goal within the practical problem solving methodology is to actually explore as many countermeasures as possible at the same time.

This concept is actually referred to as set based countermeasures and is modeled after the principles of set based concurrent engineering. Let me explain.

In traditional problem solving approaches a team may narrow things down to one countermeasure. They then set off to implement this countermeasure with sincere hope it works!

Now when we compare this to the set based countermeasures approach we see that the team has selected 4 countermeasures in the image above and begins to work to implement them at the same time.

There are many benefits to taking a set based countermeasure approach such as the fact that it builds consensus rather than having team members promote or defend their favorite solutions. It also fosters dialog and responsibility for understanding the problem more deeply while preventing decisions from being made too quickly.