*Wysowl Pty Ltd*
ACN 010 677 022

18 Miller Street
Ph: Intl+ 61 7 3359 9918
Mob: Intl+ 61 407 149 966
john@wysowl.com.au
www.wysowl.com.au

*Wysowl Pty Ltd*
*Newsletter Number 26*
*May 2010*

# HOW BAD THINGS HAPPEN
## IN ELECTRONICS R&D AND ENGINEERING

**By John McConnell**

***It is almost routine to find R&D/Engineering operations struggling with low throughput, projects delivered late and a high field failure rate. This newsletter examines some common problems and offers some potential solutions.***

**BACKGROUND**

Most readers will be familiar with the term Full Time Equivalent (FTE). It is used to describe staff levels. For the purpose of this newsletter I have created a similar measure, the Average Project Equivalent (APE). A model to show Little's Law in action in R&D/Engineering was created, originally using the more familiar measure of engineering hours required. However, it was difficult to get all three of the terms of Little's Law on a single graph where the scale allowed the changes to all three to be easily visible. That is the reason for using the measure of APE. One can easily see the outcomes on a graph.

It is essential to understand the fundamental relationships that exist between Throughput Volume (TV), Work-In-Progress (WIP) and Cycle Time (CT) which are described by Little's Law, as follows:(1) (2)

Ave TV = Ave WIP / Ave CT    or    Ave CT = Ave WIP / Ave TV

An important corollary to Little's law is that as variation increases, so too does CT. One need not be possessed of a giant intellect to see that if we reduce and control variation and CT, we control everything. If we reduce CT, we have the option of increasing TV, decreasing WIP, or some combination of both.

**SETTING UP THE MODEL**

First a simple model was created in a spreadsheet. A perfect R&D/Engineering business was created. An average of five APEs was launched every month, and an average of five APEs was completed each month. In both cases, these figures were allowed to randomly vary between 4 and 6 APEs per month. We start with 30 APEs in the system as WIP. The outcome of this situation can be seen at Figure 1.

**FIGURE 1**
**START POINT – A PERFECT OPERATION**

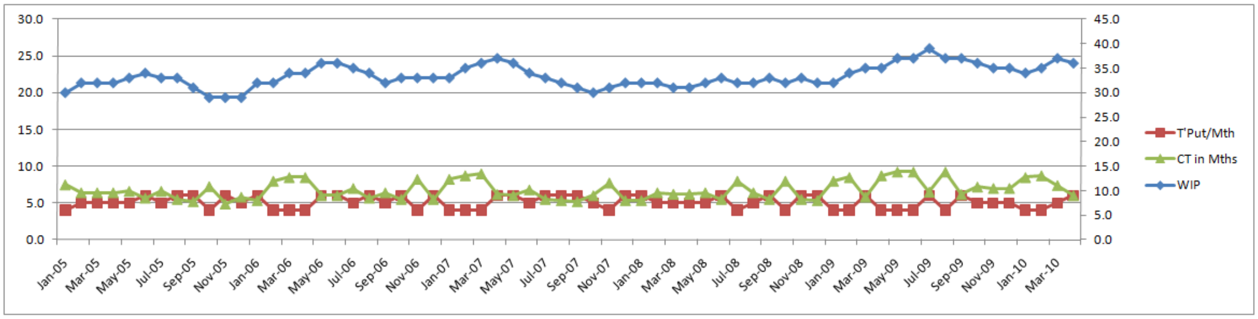
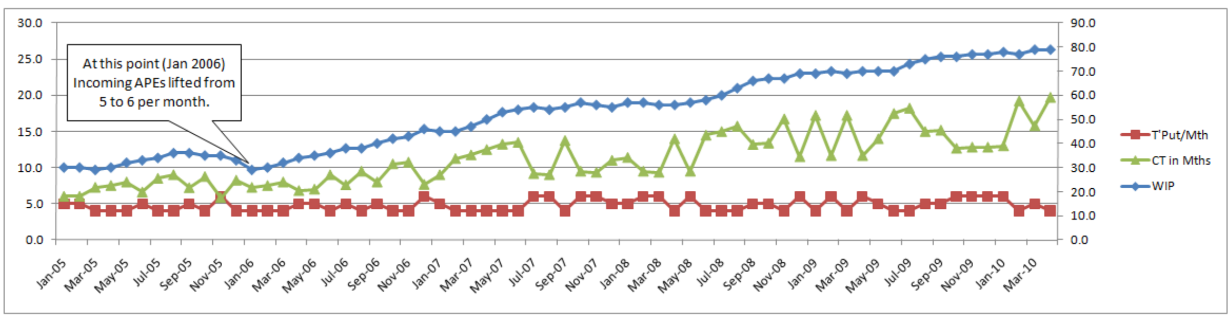


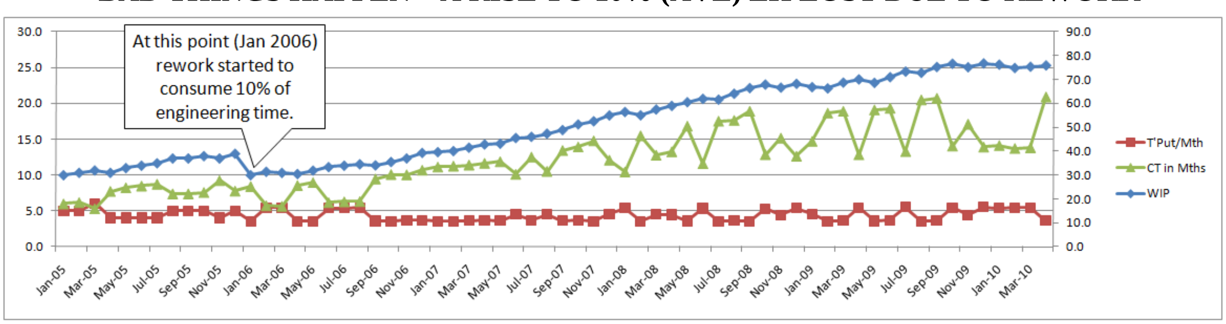

Figure 1 shows both CT and TV reasonably constant at an average of five APEs per month. WIP drifts around a little. (Like inventory, it always does. It does not randomise like Red Beads.) This situation is as close to perfect as we can imagine, which means it almost never happens. Eventually, bad things happen. In R&D/Engineering, one of those bad things is a lift in incoming work not matched by a lift in resources. The chart at Figure 2 shows what happens if the incoming APEs rise from an average of 5 to 6 per month without an attendant rise in engineering hours available. Figure 3 shows what will happen if quality is allowed to degrade so that rework is allowed to absorb 10% of the engineering hours (EH). Rework is defined as fixing “things gone wrong”, and does not include the natural, iterative nature of R&D work.

**FIGURE 2**
**BAD THINGS HAPPEN – A RISE IN INCOMING APEs**




The outcomes in Figures 2 and 3 are very similar. In Figure 1 there is no drop in TV, but both WIP and CT soar. In Figure 3, there is a 10% drop in TV, but once again, even with an unchanged arrival rate for new projects, both WIP and CT soar. In both cases this happens because there are either no buffers (spare engineers) or inadequate buffers in the system.

**FIGURE 3**
**BAD THINGS HAPPEN – A RISE TO 10% (AVE) EH LOST DUE TO REWORK**



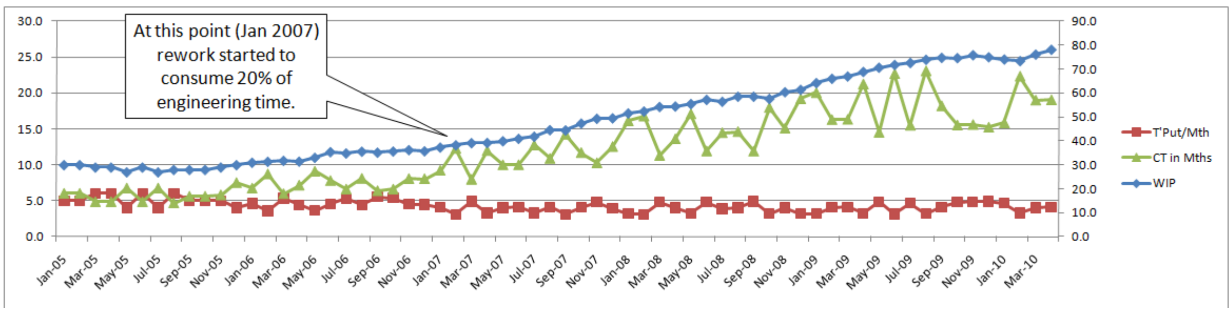
**CYCLE TIME IS THE KEY**

The situation in Figure 3, a 10% drop in TV is likely tolerable for quite a while. What is not tolerable in most businesses is the rise in CT. In both the cases at Figures 2 and 3, CT had doubled in less than a year. Customers and marketing people start complaining, scheduling becomes a nightmare, and deadlines are routinely missed. This will not be tolerated for long. In this case, at the end of 2006 significant pressure was placed on R&D/Engineering to hit their schedules...or else!

The most common resultant of such a situation is that people start to rush the work, and/or they eliminate certain integration testing to speed matters up, and/or...you see the picture. The worst thing we could do, of course, is to rush the work between VOC and functional specifications. This makes code developing and testing riskier and more difficult. The almost inevitable outcome from all the above responses is a degradation of quality with an attendant rise in rework. Rework can be done either post testing but pre-release or can be rework done as a result of field failures. Again, for our purposes here the term rework does not refer to the natural, iterative nature of R&D/Engineering.

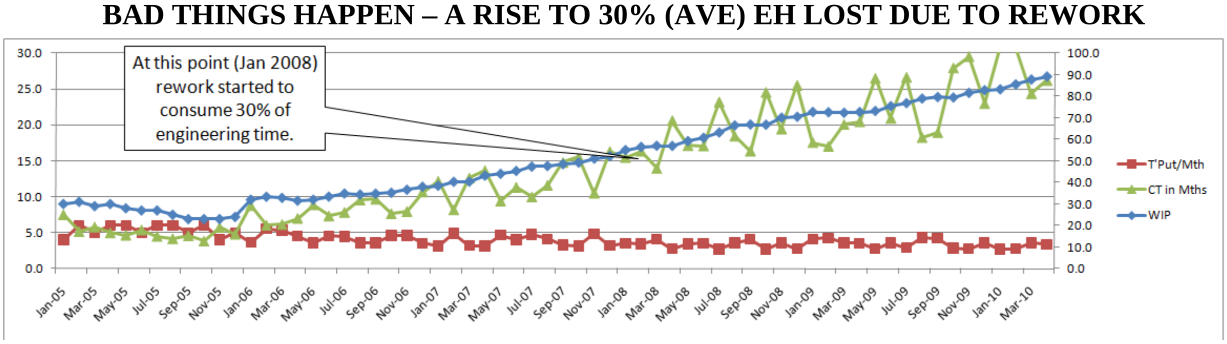
As rework rises, TV will drop. This is an issue in itself, but in many cases a bigger problem is the unexpected rise in CT. In Figure 4, the rework level noted in Figure 3 has been raised from an average of 10% to an average of 20% in Jan 2007. Everything else remains unchanged.

**FIGURE 4**
**BAD THINGS HAPPEN – A RISE TO 20% (AVE) EH LOST DUE TO REWORK**




By Jan 2008, CT has soared again to 16 months. CT has more than tripled in the space of two years. To make matters worse, TV has dropped again, but it is only 20% lower than the original 2005 condition. Our response, like Boxer the horse from "Animal Farm" is to work harder; to drive people to meet their deadlines. More shortcuts are taken. Quality is compromised again, and "things gone wrong", including rework, rise again. In Figure 5, the model shows what happens if rework and things gone wrong consume 30% of engineering time.

**FIGURE 5**
**BAD THINGS HAPPEN – A RISE TO 30% (AVE) EH LOST DUE TO REWORK**



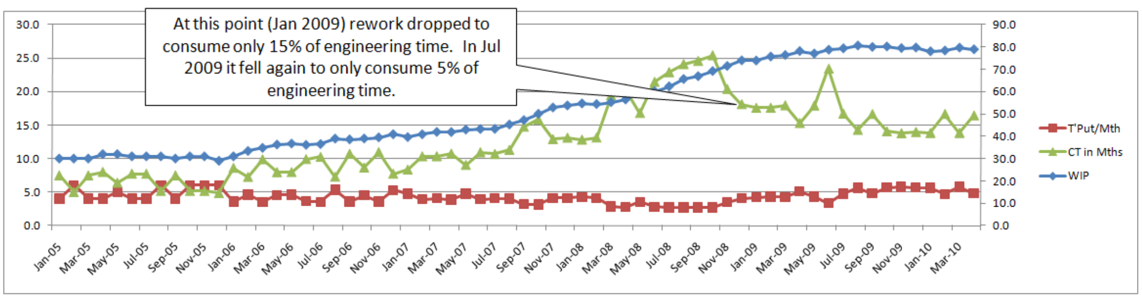
TV certainly dropped. It is now 30% lower than the initial 2005 state. However, CT has exploded. By late 2008 it has reached 25 months, five times the CT in the initial 2005 condition.

Now we seem to need a miracle to drag us out of the death spiral. All my clients have plenty of good engineers. That is not the biggest problem. The problem is an inadequate understanding of how quality impacts CT; of not understanding how a small loss of EH due to them being expended fixing/reworking things can lead to an explosion in CT, as well as reduced TV.

**TURNING AROUND AN R&D/ENGINEERING OPERATION**

Out of the mist rides a Wizard to take over the operation. Our Wizard understands that great quality begets short CT, deadlines that get routinely hit and happy customers and marketing people. In late 2008, our Wizard rolls out a single minded thrust to improve quality and to eliminate rework. He commences with the front and back ends of the process. Better VOC, improved PDD and great functional specifications prepare the future. Improved QA work ensures the field failure rates drop. In Jan 2009, the rework levels start to drop, initially to 15%. By Jul 2009, they had fallen to only 5%. The result can be seen in Figure 6.

**FIGURE 6**
**WIZADRY – A DROP TO 15% AND THEN TO 5% (AVE) EH LOST DUE TO REWORK**




TV is now almost identical to the "perfect" state for 2005. WIP has stopped climbing, and in fact is beginning to turn downwards. Most importantly, CT is coming down fast, and will continue to fall in the coming months, as more quality improvements are made.

**THE BIG MISTAKE**

A very common thinking error in R&D/Engineering is to believe that throughput and hitting schedule are pretty much the same thing. They are not. In the example given here between Figure 3 and Figure 6 the only variable that was changed was the amount of EH consumed by rework. The model allows one to adjust the number of engineers. This has not been done. It allows one to reduce effective EH as a result of "body shuffling" between projects. This has not been done. Several other variables exist. They have all been left constant. And yet this one variable had a one-off linear impact on TV and an ever increasing cumulative impact on CT. That is the big lesson. If we control CT, we control everything. If there is one variable about which we must forever be vigilant, it is CT. The biggest cause of problems with CT is "things gone wrong"; Red Beads; rework. In electronics R&D/Engineering, no level of Red Beads is acceptable. We must work now and forever to drive them to as low a level as possible, but try telling that to someone doing a cost-benefit analysis.

Recently, a manager in a client company explained that he had eliminated testing by the code developers. All testing was now to be done in QA, where it was cheaper. He accepted that more errors would enter QA, but claimed that the cost of finding them was lower, and that the engineering hours he saved by eliminating testing by engineers gave him more usable engineering hours with which to develop code, and therefore create higher throughput. Then he suggested that this higher throughput would lead to him being able to hit schedule more often.

As Deming was fond of saying: “There is no substitute for knowledge”. Readers will not be surprised to learn that this manager’s well meaning actions led to a steep climb in CT, and that hitting schedule became an impossible dream. This sounds like a good time to recall that good intentions are the stuff with which the road to Hell is paved. There is no substitute for knowledge. Eliminating the testing by developers only put more bugs into testing. Some were found during testing, and were returned to the developers as rework. Over time, rework has a much bigger impact on CT than it does on TV, but how often have any of us seen this understanding in a cost-benefit analysis? Other bugs made their way into the field before becoming apparent, after which time they reported as rework. Many months have now passed since the code was cut. The engineer who reworks the code is usually not the engineer who wrote it. The rework engineer may spend days or weeks finding and fixing a bug that the original developer would have found and remedied in a couple of hours, had she been allowed to test her own code; had she been allowed to take pride and joy in her work.

**THERE’S NOTHING NEW IN OUR WORLD**

The lessons from this simple example are not new. When our automobile manufacturers compared their product with their Japanese counterparts in the early 1980s, they discovered that the field failure rates of Japanese cars was less than a quarter than those made in the US or Australia. The Japanese tried to tell us that the secret was to improve quality, but for the most part we stuck steadfastly with our “stretch targets” and cost cutting approaches. It did not occur to most of us that the great quality of Japanese cars and the high productivity of their factories were essentially the same thing, so we kept driving people to produce more, faster, in a poor system. Our newspapers are full of the cost of that decision. Billions of taxpayers’ dollars have been spent to bail out car companies that refused to learn that the open secret to great productivity was great quality.

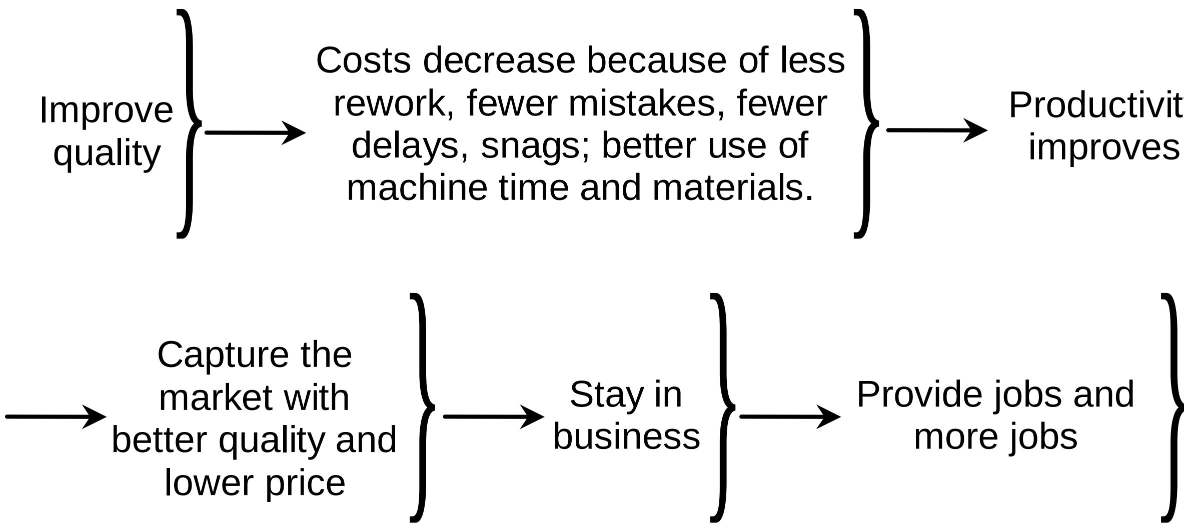
When Motorola benchmarked the *“crème de la crème”* of the electronics world in the early 1980s, they discovered that all the world class electronics outfits were characterised by low variation and brilliant quality. Field failure and rework rates were extraordinarily low. Out of this experience and the work of an engineer named Bill Smith was born what we now call Six Sigma. The Motorola folk worked out that the only acceptable solution for electronics (hardware and software) was very low variation and extraordinarily low rates of “things gone wrong”. This meant doing a lot more work at the front end of the process to properly set up the downstream steps for success.

A major problem with TQC/TQM, Kanban (Just-In-Time), Lean and now Six Sigma is that these subjects fall into the hands of consultants. In too many cases consultants either dumb the material down to make easier to teach and to sell, or they over-complicate it to maximise training hours and revenues. The concept we call Six Sigma is underpinned by a couple of simple but powerful ideas. First, reduce variation in everything you do to achieve very low levels of “things gone wrong” and brilliant First Pass Yield. Second, approach everything you do on a total process basis; what is sometimes referred to as Systems Thinking. The latter element teaches us that we get powerful leverage by working upstream. In electronics, this means doing a first class job from identifying VOC through to creating functional specifications.

The irony is that the Japanese learned the quality approach from Americans such as Dr Deming and Dr Juran. To this day the most prestigious award one can win in Japan for quality and productivity is the Deming Medal. Deming is almost a deity in Japan, but very few Americans have heard of him. Still fewer even begin to understand his work.

At the Hotel de Yama, Mount Hakone, in the summer of 1950, Deming commenced his first of many seminars in Japan by drawing on the blackboard what he called a chain reaction. It is reproduced at Figure 7.

**FIGURE 7**
**THE DEMING CHAIN REACTION**




Deming was, of course, quite correct. So too was Little.

The approach outlined in this newsletter is tried and tested and proven in electronics. Electronics ***is*** the birthplace of Six Sigma, and whilst nearly everybody in electronics is possessed of the Six Sigma buzzwords and perhaps a set of impressive statistical tools, very few people in electronics understand fully the central concepts.

Transformation is possible. It can be done. It has been done. It will be done again; but only by those who know what they must do, why they must do it, and how to do it. There is no substitute for knowledge.

**References:**


1. *Factory Physics*, W. Hopp & M. Spearman, McGraw-Hill, 1996
2. *Six Sigma in the Pharmaceutical Industry*, J. McConnell and B. Nunnally, CRC Press, 2007