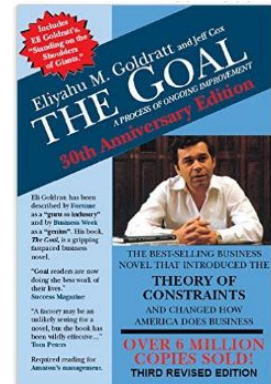


The Goal - A process of Ongoing Improvement

Plot Summary

Alex Rogo is a harried plant manager working ever more desperately to try improve performance. His factory is rapidly heading for disaster. He has ninety days to save his plant - or it will be closed by corporate HQ, with hundreds of job losses. It takes a chance meeting with a professor from student days - Jonah - to help him break out of conventional ways of thinking to see what needs to be done. The story of Alex's fight to save his plant is more than compulsive reading. It contains a serious message for all managers in industry and explains the ideas, which underline the Theory of Constraints (TOC), developed by Eli Goldratt.



What is the goal of a manufacturing organization?

Cost-effective purchasing? Employing good people? High technology? Producing products?
Producing high quality products? Selling quality products? Capturing market share?

The goal of a manufacturing organization is to make money.

Measurements which express the goal

- Throughput - the rate at which the system generates money through sales
- Inventory - all the money that the system has invested in purchasing things which it intends to sell
- Operational expense - all the money the system spends in order to turn inventory into throughput

If the goal is to make money, then in terms of the measurements the goal is to reduce operational expense and reduce inventory while simultaneously increasing throughput.

Jonah: "When you see one of your workers standing idle with nothing to do, is that good or bad for the company?"

Is inventory an asset or a liability?

Do you have a lot of excess inventory? When you lay off people, do you increase sales? Do you reduce your inventory?

Parallel: What is throughput in terms of software? What is inventory?

Two phenomena which are found in every plant

- Dependent events - an event, or series of events, must take place before another can begin
- Statistical fluctuations - most of the factors critical to running your plant successfully cannot be determined precisely ahead of time

Story - the hike - a set of dependent events in combination with statistical fluctuation. Each is fluctuating in speed, but the ability to go faster than average is restricted, it depends on all the others ahead in line. There are limits on how fast I can go. However, there is no limit on my ability to slow down. Or stop. What's happening isn't an averaging out of the fluctuations in various speeds, but an accumulation of the fluctuations, and mostly it's an accumulation of slowness because dependency limits the opportunities for higher fluctuations.

What are examples of this in your own organization? Development? QA?

Two types of resources

- Bottleneck - any resource whose capacity is equal to or less than the demand placed upon it
- Non-Bottleneck - any resource whose capacity is greater than the demand placed upon it

Which of the two types of resources determines the effective capacity of the plant?

To increase the capacity of the plant is to increase the capacity of only the bottleneck.

Story - NCX-10 n/c machine idle on lunch break, pile of inventory in front of heat treat

Jonah: "Whatever is available, the demand is even greater. If you lose one of those hours, or even half of it, you have lost it forever. You cannot recover it someplace else in the system. Your throughput for the entire plant will be lower by whatever amount the bottleneck produces in that time. And that makes an enormously expensive lunch break"

Story - Quality control

Jonah: “Make sure the bottleneck works only on good parts by weeding out the ones that are defective. If you scrap a part before it reaches the bottleneck, all you have lost is a scrapped part. But if you scrap a part after it’s passed the bottleneck, you have lost time that cannot be recovered.”

Parallel: Automated Unit Testing before Manual Testing & QA

Time and Batch sizes

From the moment material comes into the plant to the minute it goes out the door as part of a finished product, you can divide that time into four elements:

- Setup - the time the part spends waiting for a resource, while the resource is preparing itself to work on the part
- Process time - the amount of time the part spends being modified into a new, more valuable form
- Queue time - the time the part spends in line for a resource while the resource is busy working on something else ahead of it
- Wait time - the time the part waits, not for a resource, but for another part so they can be assembled together

For parts that are going through bottlenecks, queue is the dominant portion because the part is stuck in front of a the bottleneck for a long time.

For parts that are only going through non-bottlenecks, wait is dominant, because they are waiting in front of assembly for parts that are coming from the bottlenecks.

If we reduce batch sizes by half, we also reduce by half the time it will take to process a batch. That means we reduce queue and wait by half as well. Reduce those by half, and we reduce by half the total time parts spend in the plant.

Reduce the time parts spend in the plant and our total lead time condenses. With faster turn-around on orders, customers get their orders faster. With shorter lead times, we can respond faster. If we can respond to the market faster, we get an advantage in the marketplace.

Process of on-going improvement

“Everywhere, improvement was interpreted as almost synonymous to cost savings. People are concentrating on reducing operating expenses as if it’s the most important measurement.”

“In the past, cost was the most important, throughput was second, and inventory was a report third.”

“Our new scale is different. Throughput is most important, then inventory - due to its impact on throughput and only then, at the tail, comes operating expenses.”

We have moved from the ‘cost world’ into the ‘throughput world’.

What is the process?

- STEP 1 IDENTIFY the systems’ constraint
(bottleneck at the NCX10 machine)
- STEP 2 Decide how to EXPLOIT the constraint
(Machines should not take lunch breaks)
- STEP 3 SUBORDINATE everything else to the above decision
(Making sure everything marches to the tune of the constraint)
- STEP 4 ELEVATE the system’s constraint
(Bringing back old machines)
- STEP 5 If, in a previous step, a constraint has been broken go back to Step 1

Match Game

Need:

Box of matches

Plate or bowl

The system is intended to “process” matches. It does this by moving a quantity of match sticks out of their box, and through each of the bowls in succession. The dice determine how many matches can be moved from one bowl to the next. The dice represent the capacity of each resource, each bowl; the set of bowls are my dependent events, my stages of production. Each has exactly the same capacity as the others, but its actual yield will fluctuate somewhat. Use only one dice. This allows the fluctuation to range from one to six. So from the first bowl, I can move to the next bowls in line any quantity of matches ranging from a minimum of one to a maximum of six.

Throughput in this system is the speed at which matches come out of the last bowl. Inventory consists of the total number of matches in all of the bowls at any time.

Five bowls to play. The idea is to move as many matches as you can from your bowl to the bowl on your right. When it's your turn, you roll the die, and the number that comes up is the number of matches you can move. But you can only move as many matches as you've got in your bowl.

How many matches do you think we can move through the line each time we go through the cycle? 3.5 is the average of the 6 numbers on the dice.

Make it interesting. Let's say that everyone has a quota of 3.5 matches per turn. Anybody who does better than that, who averages more than 3.5 matches, doesn't have to wash any dishes tonight.

Record the amount that each of them deviates from the average. They all start at zero. If the roll of the die is a 4, 5, or 6 then record - respectively - a gain of .5, 1.5, or 2.5. And if the roll is a 1, 2, or 3, then record a loss of -2.5, -1.5, or -.5. The deviations have to be cumulative.

	ANDY	BEN	CHUCK	DAVE	EVAN
Turn:	1234567890	1234567890	1234567890	1234567890	1234567890
Roll - - - -	26425	46152	43225	16351	13641
#Moved	26452	26152	23225	14221	13321
Inventory:		00303	03252	10004	01000
Change +/-					
+2.5					
+2					
+1.5		*			
+1	*	*			
+0.5					
0		*			
-0.5			*		
-1					
-1.5	*	*			
-2			*	*	
-2.5				*	*
-3					*
-3.5			*	*	
-4					*
-4.5					
-5			*	*	
-5.5					*
-6					
-6.5					
-7					
-7.5				*	
-8					*
-8.5					

Notes

Questions to ask the group

Show of hands, who has read the Dev Ops Handbook?

Who has read The Phoenix Project?

Has anyone read The Goal?