

Inventory – The First Lean Waste



March, 2009

Nigel Cox, CFPIM, CIRM, MBSP
enVista enterprise solutions



Contents

Why Is Inventory the FIRST Lean Waste?	3
A History of Inventory Models	4
<i>The Operations Trade-off</i>	4
<i>The Pain Principle</i>	5
<i>Re-Order Point</i>	5
<i>Periodic Review</i>	6
<i>Two-Bin</i>	7
<i>Inventory Level</i>	7
<i>The Advent of Cost Accounting</i>	8
<i>The Pursuit of Labor Productivity</i>	9
<i>Material Requirements Planning (MRP)</i>	11
<i>Master Planning</i>	13
<i>The Lead Time Syndrome</i>	15
<i>Lean</i>	17
<i>Manufacturing Cell Design</i>	18
<i>The One-Card Kanban System</i>	19
<i>Kanban Planning</i>	23
<i>Theory of Constraints (TOC)</i>	25
<i>Six Sigma – 6σ</i>	25
Conclusions	26



Why Is Inventory the FIRST Lean Waste?

“Lean” is most broadly described as an attack on wastes. Wastes (Muda) are anything that consumes resources (and incurs costs) that does not add value. Something only qualifies as adding value if it’s concluded that the customer is prepared to pay for it. The Seven Wastes are:

1. **Defects** – wasted effort to create something the customer rejects. This forces added waste management processes (i.e., more waste).
2. **Over-processing** – using anything (materials, resources, unwanted features) that’s more expensive than needed by the customer.
3. **Over-production** – production or acquisition that is more than needed, generally to hide production problems, becomes inventory.
4. **Inventory** – raw materials, work-in-process, or finished goods. If value is not being actively added, even for a small period of time, this is waste.
5. **Conveyance** – no added value. Each time product is moved, quality is threatened and there is risk of loss. Ideally, manufacturing steps are adjacent, enhancing quality.
6. **Waiting** – time wasted by workers waiting for resources or waiting for a pull signal [covered later]. Waiting often leads to additional non-value-added processes to manage that waiting.
7. **Motion** – of the worker or equipment. Motion that does not add value is waste. There are also quality / wear / safety implications.

While some may argue HOW wasteful some of these areas may be in their enterprise, the list makes intuitive sense. So why do I suggest that inventory is the first of these wastes?

- **History:** Lean purists may resent this, but I suggest that Lean evolved from at least two prior disciplines: “zero Inventory” and then “just-in-time.” These focused on inventory and lead time reduction. I will show later how these are really two sides of the same coin.
- **Controversy:** Some may argue more forcefully about the benefits of carrying inventory, and many rely on it as their primary mechanism for enhancing or protecting customer service. People have a natural tendency to “squirrel” as the solution to supply-demand imbalances. Think about the last time you went to your kitchen cupboard and found you were out of something you wanted. Did you tell yourself you were never going to let that happen again? And how, I wonder?
- **Importance:** It could be argued that defects (i.e., quality) is a more important category. But it turns out that inventory reduction and quality improvement often go together. Businesses have had decades of “solving” quality problems by carrying “safety” stock. This, of course, doesn’t solve problems; it hides them. Lean proponents argue that removing the crutches that inventory provides exposes and forces businesses to address the real underlying problems – poor

manufacturing quality, poor supplier quality, supplier delivery performance, and even erratic customer demand (perhaps self-inflicted by fire-sales, promotions, quantity discounts), etc.

In this paper, we will focus on the Lean inventory model. There are many, many aspects of Lean we won't address here or that we will merely touch on. Great texts are available that cover the topic well. Check Lean.org for some references. We'll look at the history of inventory models leading up to Lean, as well, for added perspective.

A History of Inventory Models

The Operations Trade-off

Running a manufacturing business has often been described as a constant pursuit of balancing three factors:



In any given business at any given time, any one of these points would be where management is currently feeling the most pain. Customer service is a problem, perhaps. We're late on too many shipments. What do we do? We could carry more finished goods inventory to improve the chances it is in stock, or maybe more intermediate inventory so we can get it ready to ship faster. Perhaps we can prioritize demand better and flex the shop floor to jump on what's most urgent, interrupting scheduled work that's already set up and running. This incurs poor efficiencies. Maybe we aggressively expedite the floor and increase shop inventories. Customer service improves. But now our costs are up, our inventories are up, or both.

Let's say cost gets management's attention next. What to do? Longer runs will help make us more efficient. Great, but we'll need more work in process and more materials inventory to achieve that. The argument goes that, in the short term, you can only improve one side of the triangle at the expense of at

least one of the other two. The trick, of course, is to improve planning and/or execution control to “raise” the whole triangle. We’ll discuss various ways of doing this and associated inventory management approaches.

A hundred years ago, factories and mass production were relatively new. Smart folks were trying to make sense of it all and maximize profits by getting more and more efficient. This was generally viewed as labor efficiency – where the very invention of the factory model had made its biggest impact. Managing inventory levels was of secondary importance. Nevertheless, inventory has been debated as long as businesses have been manufacturing. When it comes to carrying inventory, how much to carry? Do I carry more as raw materials or finished goods? How much intermediate, partially manufactured, or sub-assemblies inventory should I carry? How frequently should I replenish and in what quantities? How and where do I store it?

At the heart of the need for inventory was the customer’s desire to come to your factory and leave with product, despite the fact that your lead times were lengthy. This forced businesses to at least carry finished goods stock. On top of that, those shop supervisors who were tasked with eking every bit of efficiency from their labor force weren’t going to pull that off by running their materials and parts inventory to a minimum, either.

Some inventory was in the pipeline to permit production runs and purchases in quantity – for economies of scale. Let’s call this **lot-size inventory** or pipeline inventory. The larger our batches, the more inventory there is in the pipeline. The rest was held to protect against uncertainty – primarily demand uncertainty and supply uncertainty. We’ll call that **safety stock**. Now let’s look at some of the ways factories decided on lot sizes and safety stock levels.

The Pain Principle

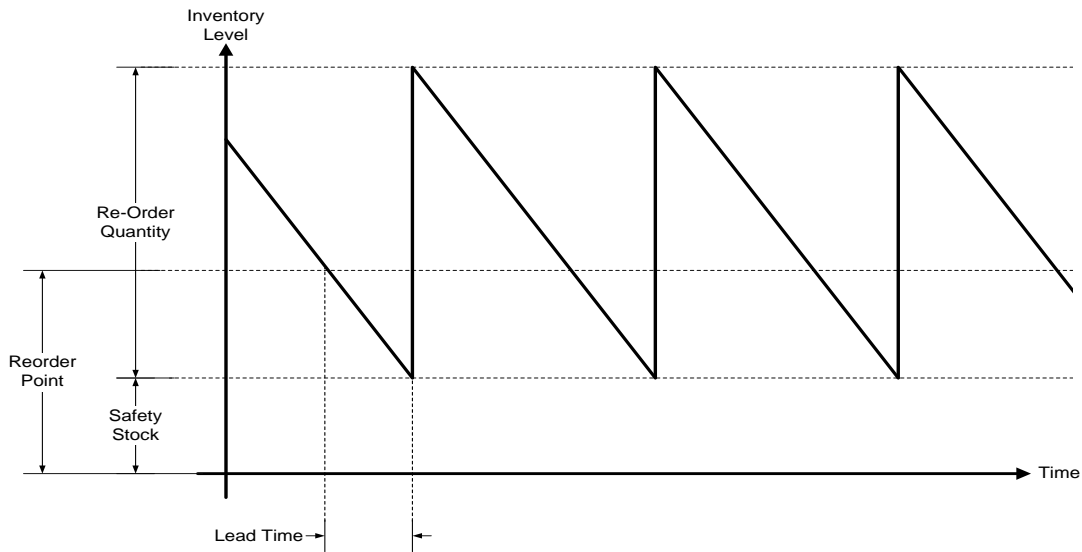
I would argue that the most common inventory management model used in history (and in place today) is “none.” You may experience some of this in your own home! I call it “managing by pain.” You carry more of whatever it was that hurt you last because you didn’t have it! This happens a lot in business, too, and in some cases, is the primary inventory model.

But let’s consider some more scientific, classic approaches that have been around for a long time.

Re-Order Point

Here’s the classic inventory ordering cycle for steady demand and steady lead times using a fixed re-order point¹ and fixed re-order **quantity**:

¹ Using the formula: Reorder Point = Usage during lead time + Safety Stock



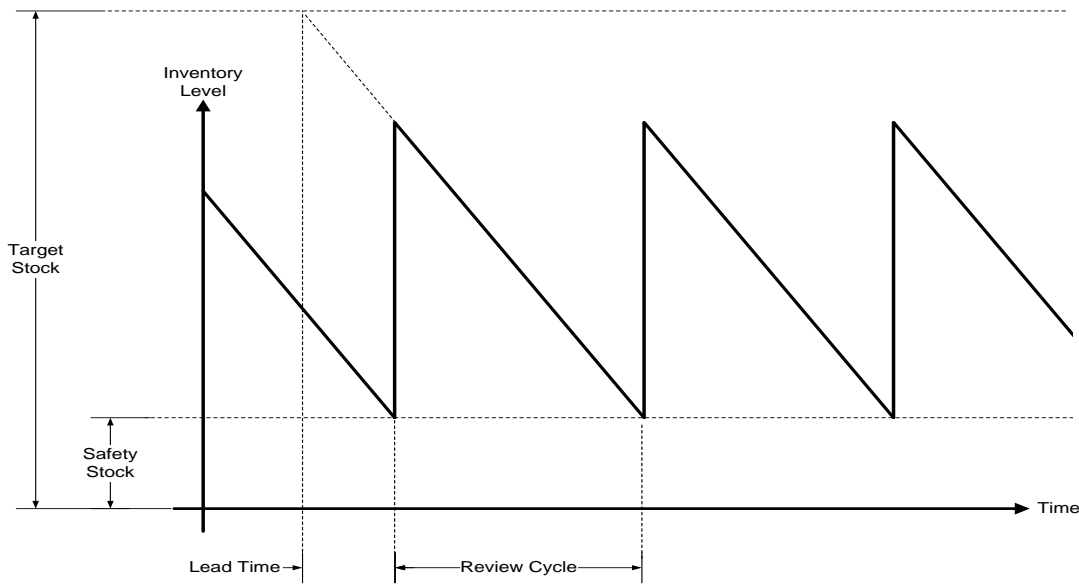
Once re-order quantities (lot sizes) had been chosen and reorder points calculated or manually set, the process became the simple act of seeing if the current stock withdrawal took the level below the re-order point. The re-ordering cycle would vary with actual demand. If, for example, demand was higher than was implied when the re-order point was set, the order cycle would shorten, but the same quantity would be ordered.

Please note: Most re-order points were infrequently recalculated with revised demand or lead times. More commonly, they would be reviewed on our famous “pain principle” -- AFTER management challenged excess / obsolete inventory levels, for example.

Periodic Review

Another classic approach was a fixed re-order **cycle** (known as periodic or cyclic review). On the same day of the week (or month for slower moving, cheaper items) an item’s inventory level would be compared to its target inventory², and the difference was placed on order. This gave the factory and vendors confidence as to when the order might arrive, if not the quantity. In steady state, it looked just like the fixed re-order point. If demand were higher than assumed here, though, the next order would occur at the same time as planned, but for a larger quantity.

² Using the formula: Target Stock = Usage during lead time + Usage during review cycle + Safety Stock



Two-Bin

A simpler variation of the re-order point approach was the two-bin system. It worked a bit like the re-order checks blank in your checkbook package. Inventory would be held in bins or cartons – often sealed. One bin might have a red tag, and would be the last one ever to open. Another might have a yellow tag. It was second to last. Unsealing the yellow tagged bin was a signal to purchase more, by sending the yellow tag to Purchasing. In some cases, the yellow tag was actually the Purchase Order itself, getting mailed to the vendor. The red tag was an expedite signal. It would go to the buyer as well. The buyer would then be on the phone to the supplier, asking when the next shipment would show up. Sounds a bit like kanbans doesn't it? This approach is still popular for cheaper, slower moving inventory that you don't mind carrying a lot of. Often a month's worth was ordered at a time.

Inventory Level

You will see from the sawtooth graphs that average lot size inventory should be one half of the re-order quantity. So ANY reduction in order quantity means a proportional reduction in average lot size inventory. So how did they pick an order quantity?

For manufacturing, this was dependent on setup and changeover costs (from one product to another). If your product was made using a casting, for example, you want to run off quite a few before the lengthy process of switching out the mold (and making a few bad ones while the process settled in). For purchasing, there was the cost of placing and later receiving and paying the purchase order. It was all manual, and quite expensive (indirect, office labor instead of direct, factory labor). In addition, your supplier was probably a manufacturer with set-up costs, and they (like you) made higher profit margins with larger orders, so they could offer discounts for larger orders. So it was (high) set-up cost versus inventory carrying cost that dictated lot sizes.

But what was the cost of carrying inventory? Mainly it was the cost of tying up money (capital) that could be invested elsewhere. Frequently, the R&D hurdle rate was used -- the rate at which an investment would have to promise return to get approved. The space dedicated to stock was a second significant cost that was sometimes also figured in. This often meant a significant understatement of the true cost of carrying inventory. Examples of other costs include:

- Insurance
- Stockroom management
- Moving materials (to make room for other materials)
- Losing inventory
- Pilferage
- Quality degradation
- Risk of obsolescence

It wasn't that early businesses weren't aware of these costs. More likely, it was that using a more complete carrying cost led to lot sizes that seemed too small and inefficient. Inventory levels can be quite emotional. The first time a shop supervisor has to send someone home because he's short parts is the time the algorithm (and carrying cost assumptions) will get challenged!

On top of this list of carrying costs is the Lean argument that inventory inherently hides other costs and blocks process and quality improvement.

The mathematically optimal lot size turns out to be where the carrying cost equaled the setup/ordering costs. Based on this, some businesses actually defined a carrying cost percentage and used a mathematical equation, involving a square root, to calculate the most "Economic Order Quantity" or EOQ. Given the assumption of level demand and fixed lead times, the math was sound. However, nobody really knew how to set the carrying cost, or they would set it too low. Frequently, it was just set to justify inventory levels considered sound, and effectively set the same high inventories "consistently" across products. The carrying cost generally became the "fudge factor" instead.

As a consequence, businesses carried more lot-size inventory and made larger batches than really was optimal. Because the costs of carrying inventory were under-estimated, they tended to carry more safety stock than they might have as well. Of course, safety stock was where the pain principle really came into play! But there were other things going on that caused a trend over decades that made things worse.

The Advent of Cost Accounting

A hundred years ago, manufacturing cost structures were dramatically different. The labor force was MUCH less productive and management overhead rates MUCH lower. Materials were relatively cheaper (if also less sophisticated). A typical cost mix might have been:

- Material 35%
- Labor 50%
- Overhead 15%

The overhead rate here is 30% (30cents of overhead for each dollar of labor). Overhead was primarily supervision and facilities. A hundred years ago, it was strongly correlated to labor. If you doubled the workforce, you needed something like double the overhead and double the space. Even the small number of office workers was largely tied to production levels – such as payroll, buying, and to an extent, accounting. Accountants couldn't determine how much overhead went to each product, so they used this convenient relationship. If labor cost \$20 for a product, they would charge (in our case) \$6 overhead to that product (by using an overhead rate of 30%). They would charge an estimate for the actual materials used in the product as well (without a burden). As a result, they had a pretty good idea of the cost. That made setting a fair-return price easy and relatively reliable. Deciding which products to grow and eliminating less profitable products was fairly easy too. At the end of the year, Accounting would compare the overhead costs accounted for by burdening product with actual overheads. If it were higher, next year's overhead rate would be increased. The accounting goal was to set an overhead rate such that the year's product output "absorbed" the year's actual overhead costs. This is the essence of Standard Costing.

The Pursuit of Labor Productivity

With this cost structure, reducing labor content was the biggest and most obvious target. Remember, Cost Accounting assumed that overhead would go down with labor. In our starting situation, we had an overhead rate of 30%. Every dollar of labor removed was assumed to remove \$1.30. Think of some ways to remove labor. The obvious one is using machines instead of hand tools and then automation – technology. At our factory, let's say we have a cost of goods sold of \$100,000 – a lot of money back then! That's \$35,000 spent on materials, \$50,000 on labor and \$15,000 on overhead each year. Spending \$8,000 for several machines, we find we can cut labor content by 5% very quickly. That should save \$3,250 a year: \$2,500 in labor and \$750 in overhead. Let's straight-line depreciate the machinery over four years, adding \$800 a year for maintenance. That's an annual cost of \$2,800 and \$450 net savings every year.

But did we really reduce overhead? This seems unlikely. The depreciation and maintenance get added to overhead, at least. The more complex manufacturing operation likely needs more management. Machines probably took longer to set up to get that efficiency – so larger lot sizes. We'll need more inventory in front of the machines, and behind them (so it will take longer before we make that product again). Remember, inventory carrying costs are part of overhead. Purchasing won't get smaller – they are still buying all the same materials and they'll be buying more "stuff" to keep the machines running. Accounting just got more complex. In sum, almost certainly, overhead goes up. At the end of the year, with less labor, we "absorbed" less overhead -- specifically, \$47,500 in labor only absorbed \$14,250 in burden. Let's say actual overhead ONLY went up by the machinery depreciation and maintenance. So it would have been \$17,800. That's \$3,550 in overhead that wasn't absorbed. Hmmm! We'd better up the

overhead rate. Keeping things flat means we need to set the overhead rate to $\$17,800/\$47,500$ or nearly 37.5% (up from 30%).

So this coming year, every dollar of direct labor we eliminate means a savings of \$1.37 instead of \$1.30. We can afford slightly MORE expensive equipment to save another 5% of the remaining labor!

Now fast forward to today. We've focused on removing labor content for decades, so it is tiny compared to a hundred years ago, and almost all that has turned into overhead. Here's a typical current-day cost structure:

- Material 50%
- Labor 10%
- Overhead 40%

First notice that the overhead rate has grown from 30% to 400% (overhead rate is typically defined in relation to labor content rather than total cost). Using the same Cost Accounting approach (and many businesses do), saving \$1 of labor implies a savings of \$5 (\$4 in overhead). Reducing labor content increases overhead rates even more today, because that overhead is less tied to labor. This magnifies the distortion that causes us to increase overhead rates the following year. Take this to its logical conclusion and the nonsense becomes clear. Eliminating labor content altogether means an overhead rate of what?

What's the relevance to inventory models? Well, traditional automation almost always meant longer runs and, thus, more inventory needed. But there are other ways of gaining labor efficiency. Ask a shop supervisor if he can improve his team's efficiency if we can guarantee he never runs out of materials, never has a shortage? Remember our operations trade-off triangle (see page 4)? Directly increasing inventory can be (and has been) seen as an investment solely to reduce labor costs.

Off-shoring the labor (outsourcing) is another way to reduce labor content. Play this through and you'll develop a similar picture to automation. Offshore some labor and the basis for overhead allocation shrinks, leading to an increase in overhead rates and, apparently, an even more attractive setting for outsourcing! In practice, these approaches rarely reduce overhead (more travel, offshore agency, management, and IT costs, for example).

Clearly, burdening labor with overhead, and thinking you will save both when labor is reduced, is flawed. Lean enterprises don't use this accounting to make investment decisions. Many enlightened businesses that are not Lean have learned to separately address direct and overhead cost savings from automation decisions. Many are now attacking overhead costs as fiercely as they did labor content. Nevertheless, we've had almost a century where we've obsessively attacked labor content, while allowing/promoting inventory levels (lot-size and safety) to be much higher than was optimal.

Some Lean advocates have suggested that inventory levels be the basis for overhead allocation, to leverage this distortion effect! Consequently, products that required higher inventory levels would have higher calculated costs³.

Labor should not have been so heavily burdened. Overhead, in modern practice, is not tied to labor content. Accounting has drastically overstated burdened labor rates, leading to inflated lot sizes. Understating carrying costs multiplied this effect further. Human nature and the drive for measured labor efficiencies probably pushed us even further to bloated lot sizes . . . that bloated for decades.

Material Requirements Planning (MRP)

MRP was the next major breakthrough in inventory management. If you've been exposed to Lean, you've probably been told that MRP is a push system and inherently means excess inventory. Yet, most MRP implementations were justified, at least in part, by the inventory reduction they could achieve. Why? Because MRP was being compared to earlier techniques like re-order point.

The big flaw in re-order point (also cyclic review, and two-bin) is that they all ignored the Bill of Materials (BOM). Re-order point is based on an estimate of demand, lead times and some carrying cost assumptions (probably implied rather than explicit). The demand was typically based on usage history rather than forecasts, especially for intermediate and raw materials. Timing of supply was dictated solely by hitting the item's re-order point – nothing predictive. For example, a component that goes into products that currently are all in excess inventory state would be blissfully “ignorant” of that fact. We could have months of finished goods piled up in our warehouse (already assembled into product) and still be ordering parts for more. It is similarly “ignorant” of its siblings (the other components that go into the same parents). If inventory is above the re-order point, don't order. If it's below, then an order needs to be in place. This is the only criterion.

IF it were thought through, safety stock would (ideally) target a “service level.” Let's say 95% available over time is our target (or for discussion purposes, what we experience). Ninety-five percent is actually fairly high, accommodating two standard deviations of variation. That means a 95% probability of checking stock and finding ENOUGH there.

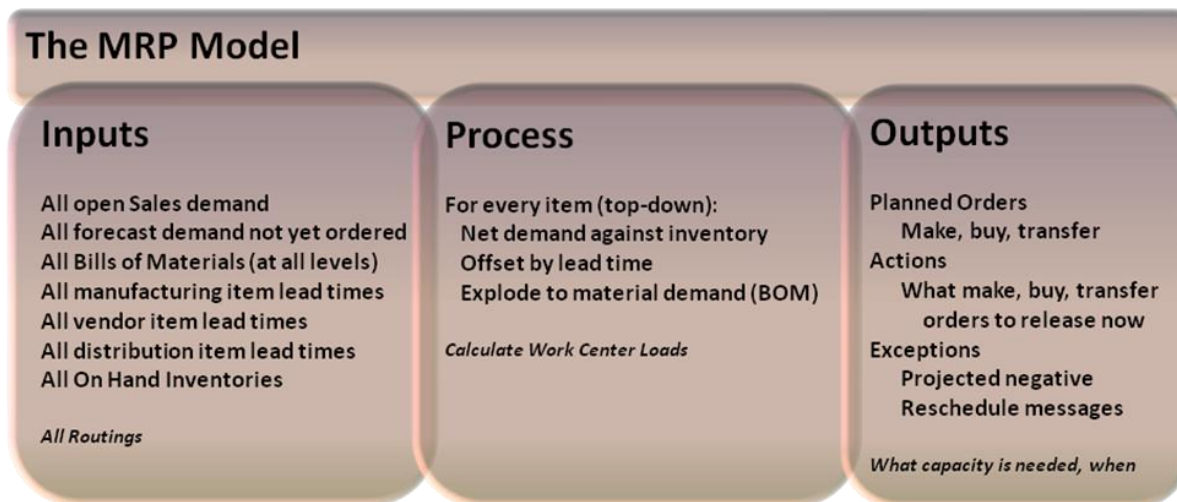
Now let's look at one of our 95% available item's parents. Let's say there are 19 other components, all also 95% available. What's the probability they are all in stock when we go to pull them? Probability theory tells us it's 0.95 to the power of 20 (0.95^{20}) or 36%! That's about a one in three chance there won't be a shortage. Another assembly might have 200 components. Now our probability that all 200 are there when we need them is 4 in 100,000! Since 200 components is not so unusual, how could manufacturers survive such odds? Several mitigating approaches were often taken:

³ Activity-based costing addresses this distortion directly, but requires an enormous data capture exercise. Value stream accounting, in Lean, attempts to bypass it altogether (by not burdening cost at the product level at all). Both can be searched on the internet if the reader is interested.

- More inventory was carried to improve the odds – especially for what we were short of last time!
- Safety stock Inventory was carried at all levels.
- Kits would be pulled early to check for shortages⁴.
- There was heavy reliance on people expediting.
- Resorting to costly expedited supply (overnight shipping, interrupting the shop floor, etc.).

Material Requirements Planning used a full BOM model and forecasted product demand, as well as open sales orders. It needed computers, where prior techniques coped with fully manual approaches. Each item’s inventory, starting with finished goods, had its future demand, period⁵ by period, netted against inventory. Time-phased net requirements would be offset by the manufacturing lead time and exploded down the BOM to become demand for the next lower level, and so on. At the end of this process⁶, there were planned supplies necessary to meet product demand specified at all levels of the BOM, given accurate starting inventory levels, accurate BOMs and accurate assumed lead times. Reams of exception notices would print out, or, these days, an interactive exceptions workbench would be populated for each materials planner. Exceptions include projected negative inventory and new orders already past due (supply needed in less than lead time), as well as reschedule notices.

There is a subtle difference here that is a plus over earlier methods. Imagine a problem with an item’s supply. It may be delaying many assemblies that use it. Let’s say 20. That’s 20 separate expedites under re-order point that could be multiple expeditors/planners. In MRP, it’s exceptions against one item from one planner. Before MRP you expedited orders (with 20 colliding expedites). Under MRP, you plan/expedite items (with 1 expedite) – if you’re a good planner. "Good" meant getting through your exceptions quickly and heading things off before they became a problem on the floor. The floor was dealing with orders. If the planner was behind, they would be barraged by expedite complaints from the floor, and MRP unravels. More than a few implementations struggled with this.



⁴ I’ve worked in facilities where a whole department was dedicated to “pre-kitting” solely to expose shortages for expediting.

⁵ A period could be anything from a day to a month.

⁶ Early MRP runs typically took several days to complete.

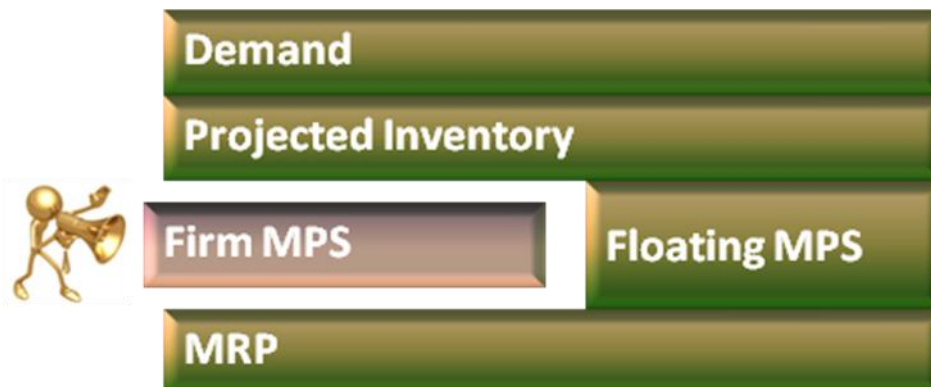
Any deviation from plan (including customer order changes, sales emerging different from forecast, or BOM/inventory corrections) would have a ripple effect down the BOMs. The most problematic deviation was sales not exactly matching forecast. Keep in mind that these forecasts were necessarily “bucketed” and the system had to decide, as sales orders were received, whether to consume the forecast and, if so, which forecast bucket. It was easy for an anticipated sales order to fall into the prior or subsequent bucket.

This created a lot of noise. One day a message would tell the planner to de-expedite a purchase order (the sales order hadn’t arrived) and the next MRP run would tell them to expedite it back (the sales order arrived now). There was also a tendency to find a solution to an exception that had a knock-on effect that wouldn’t be noticed until the next MRP run. To meet this sales order due date, say we expedite the assembly production order, scheduling to complete it in less than lead time. Next run, we get a message warning that parts won’t be available, so we schedule the production order out again. And so on.

This phenomenon was given the name “system nervousness.” One invention to mitigate this was the “firm planned order.” The planner was granted the ability to “firm” any given planned order. MRP knew not to re-plan anything “firmed.” But the major breakthrough to combat nervousness was master planning.

Master Planning

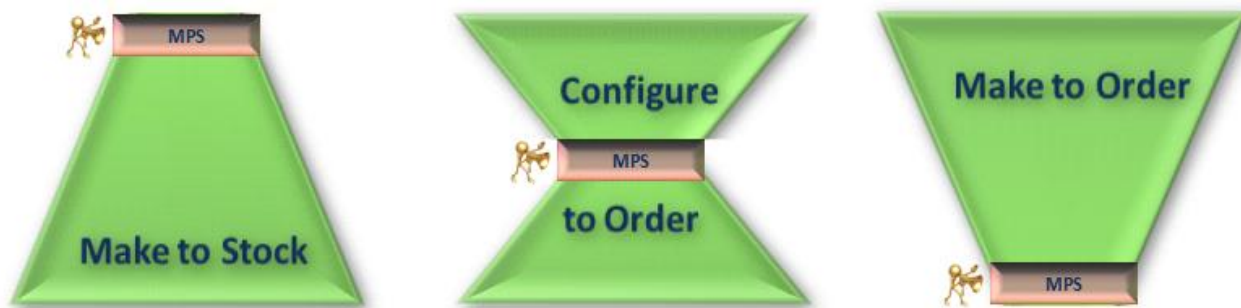
Instead of pacing the entire plan directly from sales and forecasts, a “manually controlled” layer was inserted as a wedge between demand and supply. Consider make to stock. Master planning would be at the finished goods level. Planners would look at sales, forecasts, and safety stocks and develop a finished goods supply plan. Within the master planning firm horizon, all supply orders would be treated as at least “firm” so the system would not move them.



This effectively de-coupled MRP from demand’s noise. Each planning cycle, the master planners would review exception messages caused by leaving the supply plan as it was. If the exception could be solved by using safety stock, the principle was to ignore it. After all, that is what safety stock was for! Beyond the MPS firm horizon, MRP would be tied directly to demand again, but this should be far enough out that it would not affect purchases. The net effect was that, as much as possible, the supply plan would be allowed to remain the same as it was predicted it would be during the last planning cycle, even though it

might not deviate from demand by temporary excess inventory and dipping into safety stock. Enlightened businesses measured their master planners on this period-to-period supply plan stability.

Master planners were, consequently, setting the manufacturing pace for the enterprise. It was recognized that this intense focus was costly, compared to MRP's automatic recalculation, so it was used sparingly. Invariably, MPS was imposed at one and only one level of the BOM. Inventory would also be emphasized at that level. Typically, it would be the BOM level with the least items (the narrowest point of the BOM). In our example stocked finished goods example, the manufacturer typically makes a few stocked items from a large number of parts and materials. Many consumer goods fit this model. For configure to order, the master schedule would be at the next level down where a few key sub-assemblies are made to stock and put together at the last minute in a myriad different ways. Web-ordered laptops are a good example. Make-to-order (especially materials-focused such as an oil refinery) could even be master-scheduled at the materials level.



MRP systems' biggest drawbacks were:

- They were monolithic. Large amounts of data had to be fed to it and large amounts of information came out, with significant human workloads.
- They were based on critical assumptions of accurate forecasts, accurate inventory records, accurate BOMs, and accurate lead times. Many MRP implementations began with this legacy data in serious disrepair.
- With given assumptions, MPR plans were approximately correct at the moment they were run (usually at night). But it took so long to process the output that planners were invariably looking at an out-of-date state.
- If things didn't go as planned, exception messages would go up. Item-by-item exception-by-exception, solutions still had to be devised, expedited, and communicated between the planner and shop floor (likely via a supervisor) or vendors (likely via a buyer). MRP tended to self-correct planned orders, but once they were released to the real world, corrections had to be manual.

The Lead Time Syndrome

During the MRP transformation, many experts saw a problem in the way lead time was modeled in MRP and all prior inventory models. One notable was George Plossl, one of the founders of APICS and my first MRP teacher. In George's classes, he had the students play his lead time game. My exposure to this game was not long after steel industry lead times had gradually expanded to over a year and then collapsed again, so lead time was very topical.

A class member was appointed as the vendor and the rest of us were buyers buying from him/her. Everyone, especially the vendor, was instructed to be as objective and truthful as possible. We all had a backlog of purchase orders and forecast demand in front of us. The vendors saw the same backlog, which amounted to four weeks of their capacity. We all placed our next orders, based on our forecasts/demand and were quoted four periods. We received the first shipments based on our backlogs – on time. This repeated for a few periods. George had deliberately given us each demands that, in aggregate, EXACTLY matched the vendor's capacity, so everything was running smoothly. But at a certain period, he bumped up aggregate demand by ONE UNIT. After that, everything was back to matching capacity.

Guess what happened? In that week, the vendor would quote five weeks (since they had been instructed they needed to quote the same lead time to everyone). We buyers now had a problem. We were competing with each other for on-time performance. The next week, we knew lead times were at five periods. We were forced to order two weeks' worth of our projected demand, instead of one!

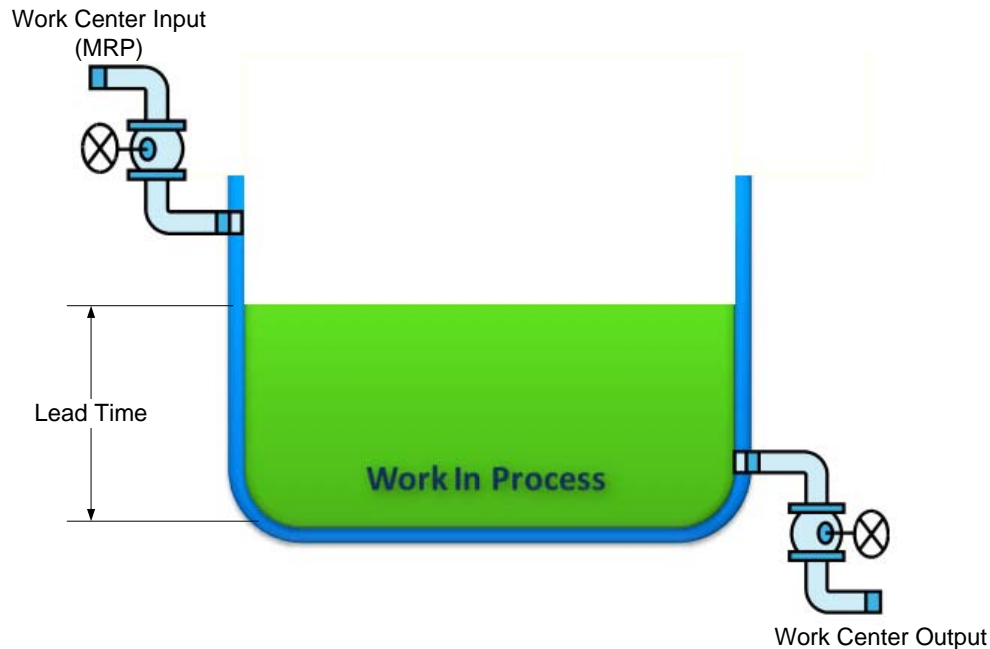
The vendor was already quoting five weeks, and everyone just placed two periods' worth of demand on them, so now the vendor had to quote six weeks' lead time. And so on. Each period we were told the lead time was a week longer! Remember, this wasn't a net overage George had imposed in every period. Our weekly demands added up to exactly the vendor's demand. That one-over in one period was having its effect on every subsequent period. In the real world, people would start to panic, buy and make things even more exponential (exactly what happened in the steel industry), but we were doing as we were told and trying to be honest.

So every period, we were told the lead time had gone out another period. By the way, we were consistently RECEIVING parts each period pretty close to our plan. Each period one buyer would be left one short. The next period that buyer might catch up and someone else would be one short. But quoted lead time was going through the roof. After about 10 cycles like this, our lead times were at 14 periods. That's when George dropped our total demand to one unit below the vendor's capacity – for one period again. The vendor was able to quote 14 periods rather than increase to 15. So we kept ordering and, before long, everyone's deliveries matched their plan, but with 14 periods lead time instead of four.

George Plossl used this game to make a few points:

- The game demonstrated that lead time and backlog are one and the same thing.
- When demand exactly matches supply, the only thing you can say about lead times is that they neither go up, nor down; they stay wherever they were. They might be really long or really short. Whichever they are, this equality says they STAY that way.

- A most important point was that when supply doesn't match demand, you shouldn't change lead times; you should change capacity. Being unwilling or unable to increase capacity was what really caused this phenomenon.
- Lead times are not inherent to the supply process; the surprising conclusion is that they are to a very large extent whatever you (the business) want them to be!



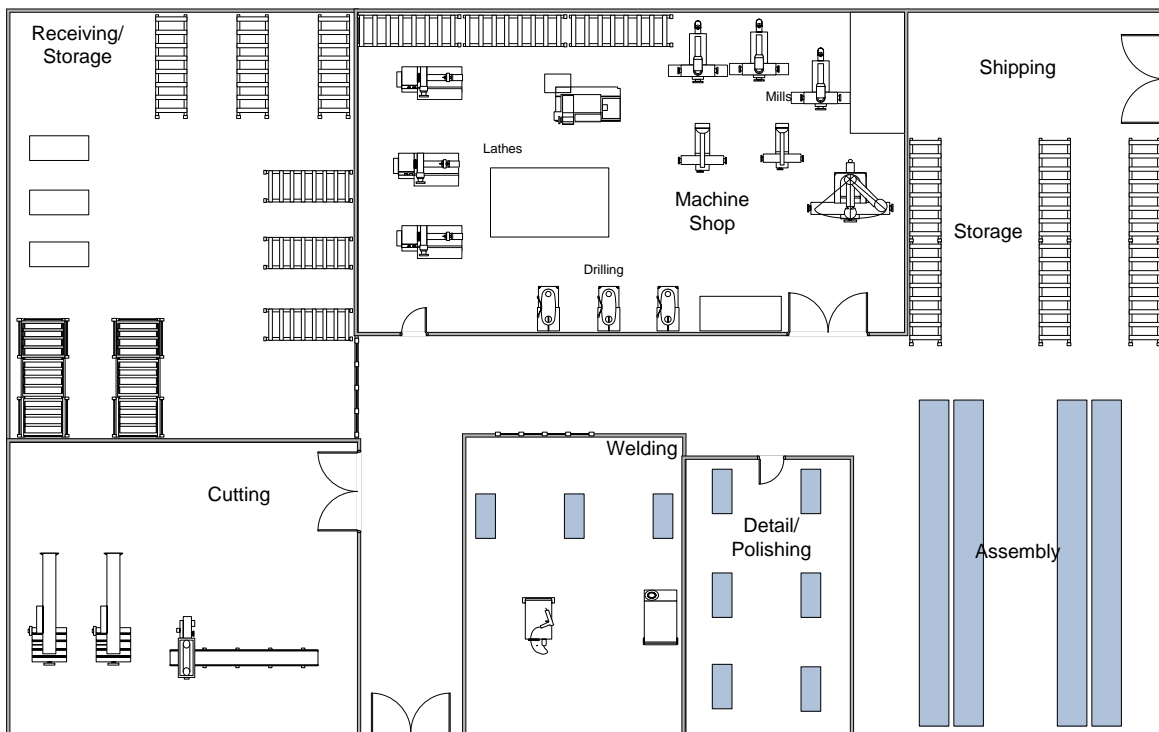
Think of a manufacturing work center as a vat, containing work in process. It has an input spigot, which is work arriving from MRP order releases or upstream work centers, on average. The amount of input is dictated by the MRP plan and, in turn, customer demand. The output is whatever the work center gets done that period. If the work center does more work than it received, the WIP level – and, thus, lead time – is reduced. If the work center does less work than it received, the WIP level and ongoing lead time increase.

The fundamental message was that lead times are not something intrinsic that you experience; they are something you manage and control to whatever level makes most sense. In theory, that level will be a function of upstream variability (including demand variability in the plan) and should be low enough that occasionally it hurts you. When it does, you try to identify the source of variability/pain (upstream quality problems, perhaps) and attempt to eliminate or reduce it.

George Plossl's recommendation was to negotiate with supervision as small a level of WIP as each work center could stand, calculate item lead times, load them into MRP, and then manage WIP levels to the negotiated level. To me, conceptually (if not historically), this was the beginning of Lean.

Lean

Today's Lean movement saw its origins in the Toyota Production System and just-in-time concept. Lean didn't build on previous techniques so much as challenge their assumptions and come up with something radically new. A non-Lean, traditional manufacturing facility is typically laid out by process. Let's take a facility that does a lot of machining and then assembles product using those machined parts. The layout might look something like this:



Traditional non-Lean Layout

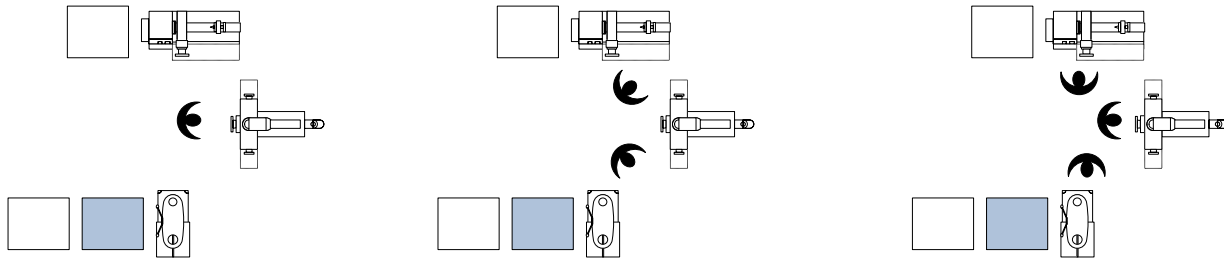
A machined part (let's imagine a gear) might get received as raw bar, cut to length, turned, milled, drilled and polished, and then held in storage until it's needed in assembly. In each of these departments, it had to be set up, so making one at a time was uneconomical. Several weeks' worth might need to be made at a time to reduce the average cost. The same is true of other parts. In addition, we don't want any machines or operators idle, so every machine has a queue of work in front of it. A machine might be making the same item for several days, so there would need to be several days of queue. In an arrangement like this, a batch of work would spend a good 80% of its time waiting. One item in the batch might be spending more than 99% of its time waiting.

Looking at materials flow, there simply wasn't good flow! It was all batch and queue for every step. And if you superimposed operation routing sequences on the layout depicted above, it would look like a bowl of spaghetti.

Under Lean, parts are analyzed and grouped into like groups. Parts in the same group would need the same kinds of operations, machines and setups. A manufacturing cell (often arranged in U shape) would

then be designed to make those like parts (let's say medium gears) by dedicating equipment and tooling. Purpose-designed jigs would be provided to minimize setup times, permitting much smaller lot sizes. Small lot size is an explicit goal in lean; indeed, single-piece flow is the ultimate target.

We might dedicate a lathe, mill, and drill to our medium gear cell. We would also design it to be operated under one, two, or three operator configurations, so that we could flex the cell's capacity to varying demand.



Medium gear cell with 1, 2, and 3 operator configurations.

The ability to flex capacity is central. Lean flipped the traditional efficiency goal on its head. Instead of making the priority to keep machines and people busy, the first priority in Lean is to keep materials busy.

Manufacturing Cell Design

Continuing with our medium gears cell, someone (we'll talk about this later) decided that four kanbans of 20 gears each was adequate for this particular gear. There are other gears made in this cell, with their own kanban definitions. Where we used to make 1,000 at a time, batching and queuing at each machine, we now process 20 at a time through all operations. There would be little or no inventory between our operations. If overall medium gear demand was low, one operator might turn a blank, switch to the miller and mill one and then switch to the drill and drill/detail one, until all 20 were done. If demand were high, we'd have three cell operators -- one operator turning, passing it to the one milling, who passes it to the one drilling/detailing.

Remember the various factors that overestimated labor costs (the burden overhead) and underestimated carrying costs and the likelihood that lot sizes have been many-fold larger than was truly optimal? Add to that Lean's frontal attack on setup times, and you can see how we were able to dramatically reduce lot sizes and save money overall through lower lot size inventories.

These days a lot of work is done to "balance" the cell, analyzing the work content so that each operation is running at the same pace (the cell's drumbeat or TAKT). Operators would at least be trained on the step upstream and the cell downstream, so they can keep things flowing, when things get out of pace. And remember, that's our goal. We want single-piece flow, not batch and queue. Ideally, operators are highly cross-trained so that they can work in a number of different cells.

We chose not to include a cut-off operation within the cell and we chose not to assemble within the cell, which would have been even better for inventory and lead time reduction. But this might not have been

practical. A goal of cell design is to have all tools and materials within easy reach. Assembly needs several machined parts, very different from one another and cell design. To machine them all and put them together could just be too complex. Several machining cells for distinct types of machined parts might feed one or more final assembly cells.

There's a lot to good cell design (search 5-S on the Web, for example), but, again, we want to focus on the impact to the inventory model.

A small buffer of inventory (pre-cut round stock in this case) sits on the inbound side of the U-shaped cell, and a small amount of finished parts might be waiting on the outbound side to be moved to a final assembly cell. In between there would be little-to-no inventory. Both would be controlled by kanbans.

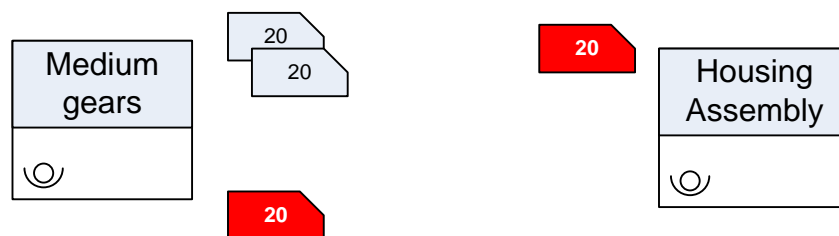
At the heart of the Toyota Production System, and the Lean inventory model, is the pull signal (or kanban). The central idea is that the act of shipping product to a customer signals permission for more of that product to be made. In more advanced implementations, it might be the act of the customer actually consuming your product that acts as the signal. It doesn't really matter what the signal is, physically. It could be an empty bin, a kanban/ticket, a fax, email, EDI message or a direct ERP link.

The One-Card⁷ Kanban System

The kanban card itself generally has an ID, specifies the item, its source (cell, vendor, warehouse, "supermarket") and its destination (cell, customer, warehouse). It will declare a quantity. Either there will be a materials list, other specifications, and work instructions (or those will be readily accessible for reference at the cell, or both).

The kanban may be paper, card, or permanently attached to a container. It could be electronic, in a computer system, or displayed on a screen at the cell. It could even be a golf ball, or a square painted on the floor. It could be a purchase kanban, a movement kanban, a production kanban, or an in-process kanban (used within the cell).

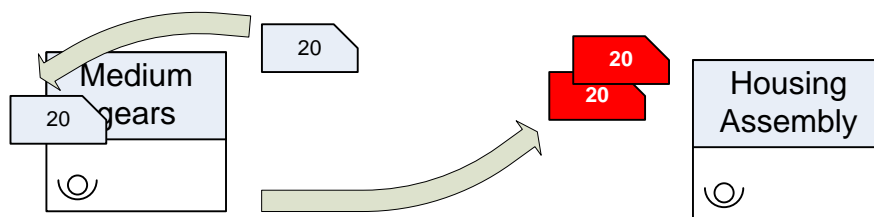
In the picture below, we show the four production kanbans for our gear. At this time, two are empty, one is full and waiting to be moved to assembly and another is full, sitting at assembly.



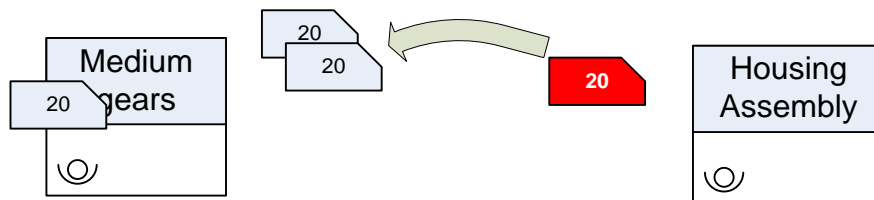
⁷ Toyota originally used a two-card system. The second card was a move ticket (to the next cell). This limited the outbound buffer at one cell independently from the inbound buffer to the next cell. The one card system limits the sum of these two buffers.

This means that our cell can produce up to 40 more of our gears, 20 at a time. We would have kanbans make other gears as well. If we completely run out of empty kanbans, the cell **STOPS WORKING**. Typically, this would set off alarms (literally, or maybe a blue flashing light at the cell), since it indicates that demand is dangerously low or there is a stoppage downstream (yes, **DOWN**stream, since we aren't getting empty kanbans sent back). A SWAT team would research, including downstream, identifying the cause. There would also be an alert if material kanbans were getting too low.

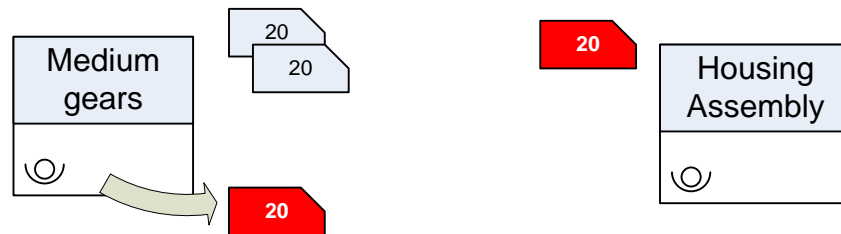
A material handler (or "water spider" in Lean vernacular) comes by on a regular cycle and moves the full kanban to the housing assembly cell, joining the one that was already there. Meanwhile, we start actually working on one of the empty kanbans:



The housing assembly cell uses the last of the 20 on its first kanban, emptying it. The water spider spots it and returns it to our medium gear cell:



Now we finish the 20 we were working on:



Get the idea? The important thing to note is that it is impossible (with disciplined operators who follow the rules) to have any more than 80 of our gears in existence. In addition, a buildup at the next higher level (housing assembly) **means** housing assembly can't make more of that kind of housing. All the kanbans for housing assembly are full. So they are not allowed to make any more and **they can't consume** our gears. Eventually, all four gear kanbans will be filled and sitting at housing assembly inbound. We don't have any empty kanbans, so we can't make more of this gear. This is how the BOM is respected, where traditional re-order point would have ignored it.

Notice that Lean BOMs aren't necessarily in an ERP System. They are institutionalized into the kanbans themselves. Given good kanban disciplines, they are inherently more accurate. Back in MRP, it was

easier to sidestep the production order and use something not specified. It's harder to get a part that isn't on a kanban. So two-bin felt like a kanban, but really wasn't. It was missing a destination.

There is a hard limit on inventory in the factory/system/pipeline this way that simply wasn't the case before. At least as important, this approach self-corrects to fluctuations in demand.

Think of mixed-model demand in an MRP world. We have four similar products with slightly different BOMs: A, B, C and D. A daily demand is forecast as even – 250 each. We run MRP nightly (weekly would be worse). MRP is starting with different on hand balances for A, B, C, and D, but planned orders reflect their demand parity over time. Net requirements are exploded through BOMs, offset by lead times. Today's recommended purchases of some of the shared components and those unique to each product are intended to meet the demand forecasted out a few weeks.

Now imagine a trended change in product mix demand. There's a run on A that's taking demand equally from B, C, and D. Our forecast still predicts an even mix, but sales orders are coming in high on A. On our next MRP run, we'll get expedite messages for A, because starting inventory will be less than was anticipated. This will also be the case for A's unique components and materials. We'll get de-expedite messages for B, C, and D and their unique components (if anyone ever gets to process de-expedite messages before the next MRP run). Nothing will actually change, unless the planner **decides** to act, by calling vendors, for example. Keep in mind that there is already a lot of inventory and WIP primed in the system -- all there on the assumption that demand would be even. In practice, the situation will get painful before anyone really changes things.

What happens with the kanban equivalent situation? To begin with, there is much less inventory in the system. Kanbans for A will start emptying faster than usual. B, C, and D will be emptying slower. Assembly will **automatically** increase the ratio of A's they build, simply because kanbans are emptying faster. So they'll be emptying their materials kanbans faster for A components and slower for the others. Let's say one of those components is our gear and it's unique to A. Because more A gear kanbans are being emptied, we'll **automatically** make more. In the same way, we'll automatically make fewer B, C, and D gears. The same applies to purchased components unique to A. The vendor will be getting more vendor kanbans. If the vendor provides similar (but different) parts for B, C, and D assemblies, all they will see is a product mix change as well. Take this a little further and imagine that one product's demand actually dries up. Hopefully, you can see that our exposure to obsolescence and excess is much less under kanban than MRP.

Take note that, in the kanban world, **no** planner, **no** planning, and **no** expediting may have been necessary. Note also that it really doesn't matter if our numbers of kanbans are off. If we have too few kanbans, we simply cycle them faster. If we have too many, they cycle slower. There's a minimum we need to keep our imperfect processes primed, but that's generally a lot less than we are operating at. The biggest exposure from inappropriate numbers of kanbans is the fact that the maximum they define is somewhat compromised.

Notice also how we are intrinsically less dependent on forecasts? It's actual demand that drives. But Lean doesn't stop there. To an extent, Lean's attack on inventory is secondary to its attack on lead times (though the two are related, as we have seen). The "nirvana" for Lean is product being created, one at a time, from scratch, in less time than the customer needs to receive it. Put another way, cumulative supply

lead time is less than customer lead time. This means that execution (making and buying) is no longer dependent on forecast at all! Few can make this claim, though.

Let's review the MRP weaknesses and compare:

- **MRP** is monolithic. Large amounts of data had to be fed to it and large amounts of information came out, with significant human workloads.

Lean: Does NOT require monolithic computer systems. Once value streams have been defined (like-products), properly mapped (using value stream mapping), wastes eliminated where possible, and cells well designed, then we generally find that we have greatly simplified processes. Good signal design means execution is controlled completely by pull signals and not planning. Planning is used for manufacturing and supply chain planning, NOT to control execution – decisions like which machine to buy, which products to promote/end of life, and what to make next.

- **MRP** runs were correct-ish at the point they were run (usually at night). It took so long to process the output that planners were invariably looking at an out-of-date state.

Lean: Self-corrects for demand fluctuation.

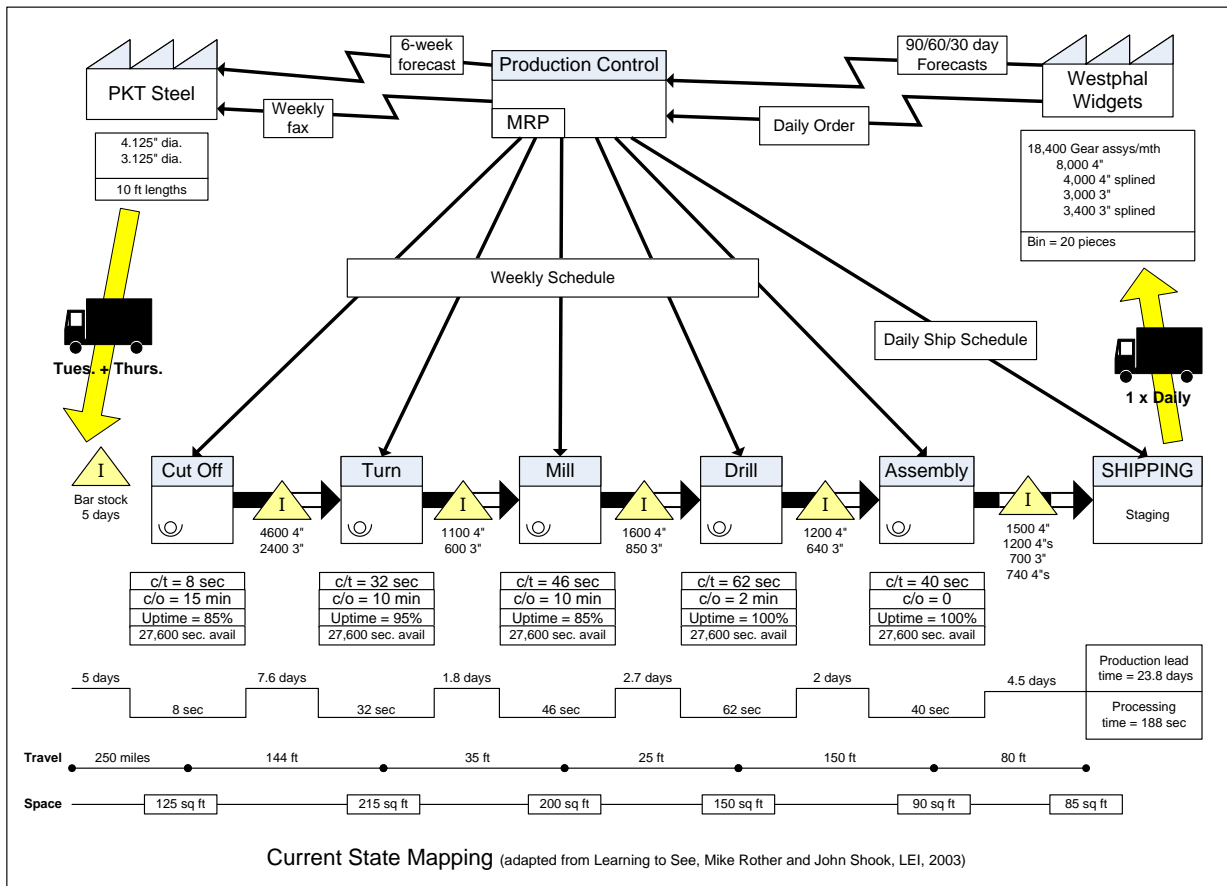
- **MRP** logic is based on critical assumptions of accurate forecasts, accurate inventory records, accurate BOMs, accurate lead times.

Lean: Directly attacks dependence on forecasts by shortening lead times and driving supply directly from actual demand.

- **MRP:** If things didn't go to plan, exception messages would go up. Solutions still had to be devised, expedited, and communicated between the planner and shop floor (likely via a supervisor) or vendors (likely via a buyer). MRP tended to self-correct planned orders, but once they were released to the real world, corrections had to be manual.

Lean: Execution does not depend on planning and self-corrects for demand fluctuation.

Kanban Planning



A new Lean initiative should start with a large characterization effort called **Value Stream Mapping**. This begins with selecting a team (shop-floor heavy), training and then **Current State Mapping** (example map above).

- **Define scope;** enterprise (across sites), factory, or process level.
- **Group like products into families to define value streams:** Pick one of these. Analyze customer demand to specify a TAKT time or drumbeat: How many must be produced each day to meet demand?
- **Map overall process flow:** What are the value-added steps needed and the non-value-added steps?
- **Identify key metrics** such as number of operators, operator cycle time, value-added time, changeover time, distance travelled, rated capacity, batch size, and others.
- **Complete process blocks.** Update the process blocks in the flow with their key metrics.
- **Draw inventory and calculate days of supply.** How does product flow from one operation to another?

- **Fill in timeline.** Show value-add durations and non-value-added durations. In the example above, the total value-added time is 188 seconds out of 23.8 days. This ratio is not untypical.
- **Draw the Raw Material Supply.** This is vendor supply.
- **Draw Information Flow.** In the example above, production control is the center of information flow.
- **Calculate the Value-Added Percentage.** In the example above, it's less than 0.01%.

At this stage, using Lean techniques, the team looks to remove all non-value-added steps that they can. Then they start designing cells, and defining kanban sizes and quantities, etc. The inventory model and plan we are interested in is really defined by the kanban sizes and quantities. Fundamentally, setting and revising these levels is Kanban Planning. Remember though, getting this a little wrong is OK. The speed at which kanbans get cycled is dictated by actual customer demand.

Kanban Size. This is the quantity on one kanban and is the equivalent of lot size. Ideally, it would be one but, in practice, should be influenced by setup or changeover time. Using an unburdened labor/machine cost and a more complete carrying cost wouldn't be a bad way of developing this. However, the quantity should be a convenient tote quantity – the number that fit into a standard bin, perhaps. Ideally, this quantity wouldn't change. We'd manipulate the number of kanbans instead.

Number of Kanbans: Done properly, this should take into account:

- **End-of-Life:** Products being withdrawn and their BOMs need to be analyzed to see which kanbans should be reduced or eliminated.
- **Engineering Change:** Components being removed or added to BOMs also need to be analyzed to see which kanbans should be added, reduced or eliminated.
- **Demand:** This is where we would want to use a forecast or component demand derived from forecasts. Some keep their MRP system solely for planning (not execution) and analyze kanban quantities against that.
- **Demand Variability:** The more variable the demand, the more kanbans needed. Naturally, we'd be innovating to reduce that variability, though.
- **Supply Variability:** The more variable the supply (production or vendor), the more kanbans needed. We need to be innovating to reduce this variability, as well.
- **Cycle Variability across Products:** Carefully placed buffers can help mitigate differences between products in a cell or from cell to cell. Let's say the second operation in a cell isn't always needed. Maintaining a small kanban buffer there can compensate. This would be an in-process kanban.

So you can get fairly scientific about kanban sizing. But at the end of the day, simply having a goal of reducing them can be an effective alternative to the science. If a product is running through a cell really

smoothly, it may be time to take one or two kanbans out. In fact, cut kanbans until it hurts and see if you can solve what emerges by team-driven process/quality improvements!

Lean planning really amounts to deciding, kanban by kanban, a “good enough” quantity. We also need to keep TAKT times up to date (i.e., demand rates) and ensure that the cells are designed accordingly.

Theory of Constraints (TOC)

Eli Goldratt (author of *The Goal*) and the Goldratt Institute promotes a distinct approach called Theory of Constraints. His argument is that some constraint (perhaps the capacity of a critical work center) is THE constraint for an enterprise. His thesis is that this constraint should be elevated and exploited. Matching the PACE of the constraint to demand paces the entire business. The constraint is managed closely and now management can somewhat take their eyes off other resources (subordinating and tying their pace to the constraint).

At some point, the constraint can be broken, when it moves somewhere else and we identify it to repeat the exploitation.

As far as an inventory model, TOC doesn't really conflict with Lean. In fact, kanbans can be the mechanism to link upstream and downstream operations from the constraint to customer demand and pace to the constraint's drumbeat. It is critical to keep the constraint busy. One approach is to have proportionally higher buffers of inventory in front of the constraint. This may not be so much carrying more inventory overall as placing it strategically. In Lean, this amounts to another consideration for kanban sizes and quantities.

Again, there's a lot more to TOC, but we're looking only at the implications on inventory models.

Six Sigma – 6 σ

Six Sigma concepts have a quality focus on reducing variability. They stand for six standard deviations, which means defects measured in parts per million. It doesn't really have an inventory model, as such. However, variability is a key driver pressuring businesses to carry more inventory. If we could eliminate variability of demand, supply, manufacturing processes, inventory accuracy and BOM accuracy, we would need very little inventory! Consider applying cause and effect, and then statistical process control to inventory accuracy. Use cycle counts not just to measure accuracy, but to see where the process of maintaining accurate inventory is “out of control.”

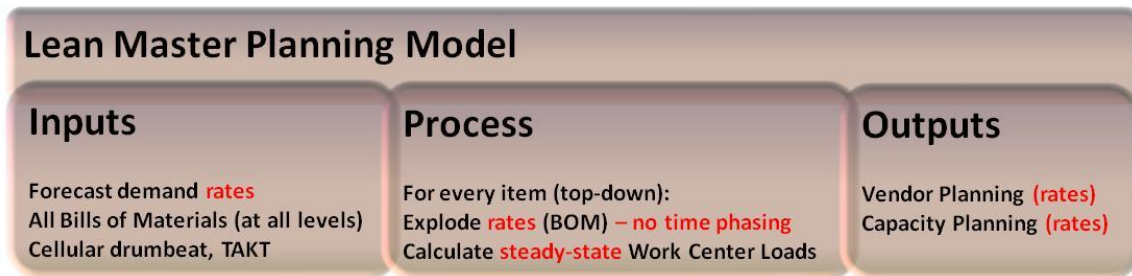
Reduction of variability is in harmony with lean. We want flow. We abhor interruptions and surprises. As we reduce variability, it's another opportunity to pull kanbans from the system and further reduce inventory and lead times. This synergy is evidenced by the fact that there are now many published texts with “Lean Six Sigma” in their title.

Conclusions

- **Too much inventory:** People naturally tend toward building inventory (and thereby extending lead times) to solve problems, especially when their inventory management and associated model aren't implemented with discipline. But this doesn't really solve problems; it hides them. Left unchecked, a business' inventory simply grows to the point where the excess and obsolete content can't be ignored anymore. Inventory is purged and blood is let and we start over building it up again. Meanwhile, our quality, supply and process problems aren't being addressed. At the end of the day, this is a huge opportunity for the competition.
- **Old models aren't all bad.** Traditional inventory models themselves were not bad for their times. Their implementation may have been flawed and they may have been constrained by limited information processing capability. All these techniques are still in use today and it's a mistake to reject them out of hand. They still have their place. Some of the best Lean implementations do not use a 100% Lean inventory model, but have focused lean approaches on the most repetitive, costly value streams.
- **Was MRP too much too fast?** In hindsight, it can be argued that we went too far, too soon with MRP. Maybe we were a little entranced by all that unleashed computer power that was put into our hands. Think about what we did. We said, "Let's model our entire supply-demand system, plug in demand and let it tell us what we should do next." Nobody thinks we can plan and schedule on "automatic." We still need people to interpret the plan and manage the enormous number of plan exceptions that spew out. But in the end, putting a big black box between the person and the problem hinders clarity, just like large inventories do. We need to get closer to the problems and cast light on them, not invest heavily on insulating ourselves from them.
- **MRP's shortcomings:** Our MRP models had two major drawbacks. First, they depended on assumptions with gaps that, in many organizations, you could drive a truck through. Accurate forecasts, inventories, BOMs and lead times to name just three. Secondly, they tended to model manufacturing execution as it was – non-cellular, complex, with built-in needs for unnecessary and too numerous piles of work in process. It's quite possible (and not unusual) to enshrine this complexity inside the MRP model and actually perpetuate it.
- **Lean simplifies execution and reduces the need for planning.** The Lean mantra is to simplify execution to the point that planning isn't needed to execute. Central to that simplification is all-out war on inventory, lead times and variability in our processes and supplies. Shop floor people who own the processes are put in charge of continuously improving on these fronts and the rest of us are there to support, facilitate and mentor.
- **Too much transaction data capture and too little process improvement:** An important corollary to Lean execution is the dramatically reduced need for data capture. Lean focuses on the processes, not the material movements. Kanbans can manage those without data capture. Process monitoring is where Lean wants its analysis effort and where statistical methods and data capture have tended to flourish. It's perfectly feasible, with good pull signals, for the business to take a four-wall approach to inventory tracking. What came into the factory and what

went out? Who cares about what happened in between if you have confidence in your processes and there's not much of it?

- Lean execution together with master planning:** In recent times we've seen many Lean enterprises embrace MRP again. This makes some sense. Once we've simplified execution to the point that it can look after itself largely without planning, we can reflect that simplicity in our planning engines and use them for what they are good at: long- and medium-range planning – not scheduling. This is an opportunity to simplify MRP as well. For longer range planning, we are less interested in projecting every little inventory movement between now and then. We are more interested in steady state rates of supply/demand. And in steady state, supply equals demand! Two of our thorns – inaccurate lead times and inaccurate inventories – disappear. Our planning engine is driven solely by forecasts, BOMs and routings (much simpler routings in Lean!). This needs a gross explosion of rates, instead of a net against projected inventory, offset, explode by time phase -- much simpler and faster data processing.



If you are new to Lean, I can't expect you to master it (become a Sensei) in 27 pages. However, you should now have a better idea of what Lean is, what preceded it and why Lean might be better than what preceded it. If you already have experience with Lean, much of this is likely familiar, but hopefully this text provided some new or fresh perspectives.

For more information, contact enVista: www.envistacorp.com, or 877-684-7700