

WMS: Is it just a game of Whack-A-Mole?

**By Robert Kennedy,
Vice President,
Business Development**

Whack-A-Mole. Such a fun game. We've all played it at some point right? You take a small bat and hit the heads of the mechanical mole when it pops up out of its hole. It's a sort of metaphor for life isn't it? Solving one problem can create another. Solve that one and a third pops up. Act too slowly and you have moles popping up all over. You can never really win the game because there is always an-



other mole. Success can only be defined by how quickly you can react to the next pop-up; by how you manage the moles. Imagine if you knew, after slamming the first mole, exactly where the next one will pop-up? And the one after that? Things would be a whole lot more manageable wouldn't they? If you played enough, you probably could get a feel for a sequence. They aren't random, they following a pat-

tern. Learn the pattern and you master the game.

Think of Whack-A-Mole now as a metaphor for a WMS start-up. Beat down one issue, and another pops up. Solve that one, a third rises to the surface, and on and on. Do it enough times and you come to realize that with a WMS implementation the sequence of issues IS predictable. It does follow a pattern. The same issues surface every time in pretty much the same sequence regardless of how well the project has been run. The only difference is the timing and severity of the issues. Every WMS implementation I have been involved, and those number over 100, followed a very similar, very predictable pattern. Think about how much more effective you can be if you know what's coming.

THE PROJECT

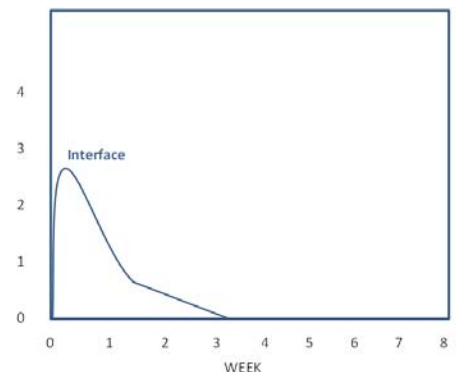
Let's build up to the day the system goes live. Your project team has just spent the last 6, maybe 9, maybe more months of preparing the system and the operation. You've been through a Requirements Process. You have identified gaps and authorized enhancements to be built. You have designed how the WMS is going to integrate with other systems, namely ERP and WCS systems, and the work on that has been completed. You have configured the system to support the process you've designed. Your data has been loaded. You've designed and executed on

a test plan. You've designed and executed on a training plan and all your users have been certified. This entire process has been supported by the best and brightest from your team and from your vendor's team. Maybe you've had help from a third party integrator or consultant. As few as a dozen people have contributed to this project, and likely, the number exceeds 50.

If you stop and consider the work that's been completed just to get you to this point, you have to consider that even if everything was managed brilliantly, the sheer volume of decisions, data, software code and hardware, coupled with the variability of talents, skills and experience of the people involved, creates a situation that statistically ensures error. Maybe the NASA could pull it off, but you don't have their budget.

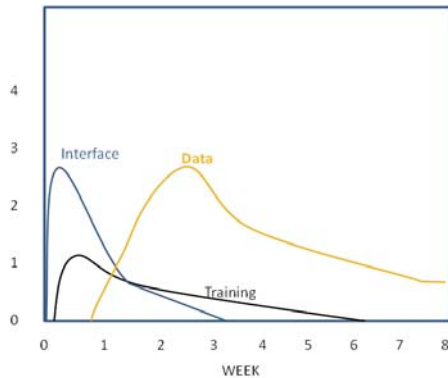
So, what can you expect on Day 1?

DAY 1



At the point of actual conversion of the old system to the new, you will already have moved the new system into a production environment. You will have a fully loaded data base and a completely configured system. For everything to go exactly right means have to you be successful in preparing thousands if not millions of data elements with thousands of combinations and paths the application can take. On top of all that potential for error, you will be using for the first time in production a set of integration software and tools that you likely have never used before. So on Day 1, you will have integration issues with the host (ERP) system. You will have issues with the hardware, and with the physical network of RF controllers and printer ports and PC's not working properly. You will have to manage various issues with the material handling systems, either with the WMS and WCS interface, or with physical issues with the conveyor.

These issues are usually fairly easy to solve, and for the most part can be rectified or worked around in a matter of few hours or a day. Worst case, by the end of the week you are receiving the data you need in the volume you need to support.



WEEK 1

After you get your systems talking to each other, and you start processing receipts and orders, you will run hot with several simultaneous issues that will impede your ability to ramp up volume. First, you will continue to have data and configuration issues. These will more likely come from wrong data, or wrong configuration as opposed to missing information. The severity of these types of issues can vary widely. It's often true that testing programs are great at testing the software but don't do a great job of testing data. It's also often true espe-

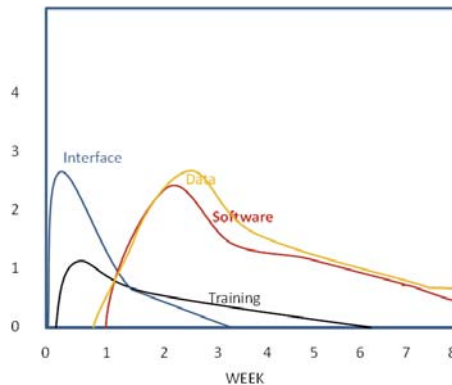
cially in converted operations (as opposed to new operations) that the existing data is not as accurate as everyone thought. Consider that its likely that the new WMS is much more sensitive to data than what its replacing, so what was good enough for the old system may cause the new system to choke.

Even as you fix the data issues, you will experience learning curve issues with the staff. Those first couple of days are like the difference between going to the driving range versus swinging your club at an actual tee - when everyone is looking and it counts the pressure is on. Your staff knows what to do, but they will be hesitant and will make simple mistakes.

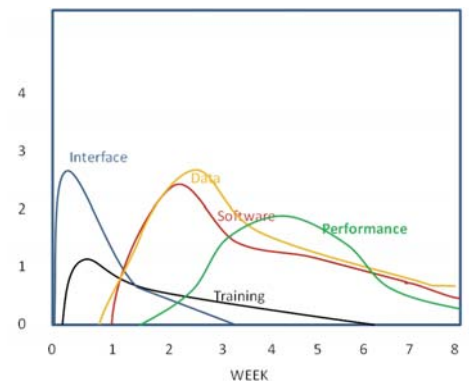
They will also encounter issues that just weren't part of the training. Regardless of the quality of training, you will still experience self-induced problems by workers doing the wrong thing or worse, trying to fix something they did wrong. We are all only human right? It's amazing sometimes the holes that workers can dig themselves into. And, they may not be comfortable with all of the exception procedures, because, likely, you didn't spend as much time on these in the class. Why would you, they are exceptions!

WEEK 2

As you move through the data issues and as your workforce gains their confidence, volumes will begin to ramp up and you will expand the scope of what you are accomplishing - more suppliers are being turned on, customers are being added. If you haven't already seen them in Week 1, software issues will start to raise their head. These are issues that will come primarily from enhancements, custom scripts or with third party systems. It's the new software that causes most of the problems. That makes sense when you



consider that the core application has been in use for some time and has been battle hardened by the experience of dozens of other companies. These can be the hardest to fix. These can take the longest to fix. Software problems have to go back to the developer for fixes, and have to run through the standard quality process. Making software changes outside of the quality process, or even on the fly on site, is never a good idea. It introduces too much risk that the new changes or fixes will cause other problems - worse problems. As such, you will find yourself designing and using workarounds that will take longer and require more people. Your throughput will stall until the fixes come through.



WEEK 3

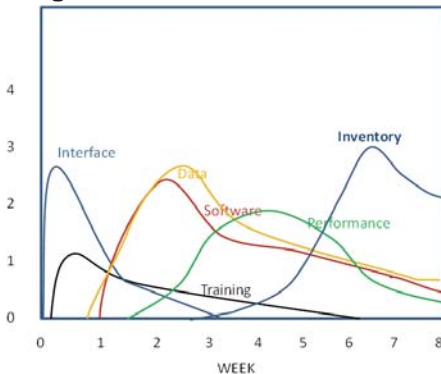
As the software stabilizes and even more volume is released, if you are going to experience performance issues they will happen about now. Is the database sized correctly? Has it been optimized? Are their specific pieces of code that are cumbersome and slowly down the system? Have you tripped a threshold with traffic from mobile devices? Now, your interfaces are humming, the workers aren't thumbing the keypads as much, your data is under control (except for the new stuff that keeps coming but you are managing that) and your software is getting stable. With these conditions, you are confident to crank up the volume to climb the approach to full production. That's when you will you start to see performance issues. The database needs tuned, your hardware needs to be reconfigured, you need to make adjustments to RF traffic, and the software itself may need to be tweaked.

Performance issues will drag an operation to its knees, creating frustration and further hurting the confidence you are trying to build in the system. These issues tend

to rear up fast, almost like tripping a wire, and can be difficult to fix.

WEEK 4

As you work your way through ALL of these waves of issues, eventually the problems they cause will manifest themselves as inventory accuracy issues. All the mispicks, all the unidentified cartons, the misreads, the bad data etc will create inventory issues. Aside from the financial impacts and business interruptions, inventory problems will also cause confidence problems with the system. It may be time to consider a cycle count initiative, at least with the picklines. Also, whenever an inventory issue arises in the warehouse, its not immediately identifiable as such, so the reflex reaction from the staff is that it's a system problem. It doesn't matter what the ACUTAL issue is. As far as they are concerned – it's a software problem. So, inventory issues have to be identified fast and to minimize their impact and to maintain confidence in the system as you work through things.



The challenge layered over attempting to fix all of these issues is the pressure you begin to feel from management as the time it takes to get to full production, and to get the savings promised, begins to lengthen.

Minimize the Impact

Everything described above will happen to you. The really good projects will see smaller impacts of less duration. The others will see trouble. On Day 1, the difference between a good and a bad project ultimately comes down to the quality of the effort you have already put in on:

1. Minimizing the introduction of new software. The more new software you have in place, the more risk you run of something not working.

2. Data validation. How much have you tested the data?

3. Testing. How thorough is your testing plan? How well did you execute? Did you test performance? Did you test the network and the hardware?

4. Training. How well did you train your staff in all that they will be expected to do? How much of the training is hands on?

Even if you are able to confidently place a checkmark next to each item, on that Day 1 you will still see the problems described. The key to success in managing these problems in production is the ability to recognize the cause and sequence, and to be prepared to rapidly address each one as they show up.

Whack-a -Mole.

A good start-up plan includes a support deployment plan. A deployment plan is designed to provide users with an identified "goto" resource to address problems quickly, with a clear path to escalation all the way back to the responsible party, which could include the vendor. The key to minimizing the issues is a quick response, which requires having the right resources at the ready when you can expect them to be needed. On Day 1, that means having the team who designed

and coded the interfaces to be engaged as the systems turn on. It means having a quick response protocol in place with the software vendor to react to software issues. It also means having an onsite deployment team to address issues that arise from the operation.

The deployment team needs to be organized in a hierarchy, with the first line using consisting of trained leadership staff; team leads or supervisors, deployed to their responsible areas on the floor. They too need to know who their escalation resource is and just as important, they need a clear understanding of how or where to contact that person. The second line should consist of superuser resources, either from the customer, vendor, or from a consulting firm. These are resources with start-up experience and a deep understanding of the application and the process it's supposed to support. Issues that need to be escalated from them need to go to the "war room" or command center, comprised of project leadership, operations leadership, vendor leadership and IT leadership.

As the start-up progresses; as the learning curve fades, and as the system stabilizes, the deployment team can be adjusted to draw down support personnel, or redeployed them to areas needing additional help.

About the Author

For more than 30 years, Bob Kennedy has planned, developed and implemented leading-edge supply chain execution systems around the globe. Bob has led more than 100 large-scale implementations of WMS in more than 15 countries and a variety of industries. He has earned an industry reputation for quality and a track record of successes. Bob is a partner at DMLogic, a team of WMS experts who provide services, support and WMS products. He is a member of MHI and CSCMP, and is a frequent presenter at conferences including ProMat, MODEX and WERC and CSCMP. Bob has a BS in Industrial Management from Carnegie-Mellon University.



DMLogic LLC

661 Andersen Drive, Foster Plaza 7
Pittsburgh, PA 15220
412.458.4010 sales@dmlogicllc.com
www.dmlogicllc.com

