



STUPID SOFTWARE

Stop blaming inanimate tools and commit to effective work management practices

While I was talking with a maintenance manager, a maintenance shop supervisor walked by, mumbling something about “stupid software.” Apparently the computerized maintenance management system (CMMS) did not meet the guy’s expectations. What was wrong with it? Did the software lock up? Did the software kick him off unexpectedly?

Apparently the company’s corporate office decided which software would be deployed to all plants. They had been sold the software with the understanding that the software vendor’s work management workflow template could be deployed to all the plants. The plants were of various sizes with various organizational structures and production requirements. It’s a common approach companies use to minimize costs, and software vendors agree with it to make the sale.

The plant managers and maintenance managers were told that the software was being implemented; it was non-negotiable. When it came time to install the software, project teams were formed at each plant; the plant manager and maintenance manager selected people closest to the work to be on the implementation project team. This resulted in a somewhat hands-off approach to show confidence in the project team; besides, they had 10 other projects that needed their attention.

The project teams included craftsmen, supervisors, and store-room representatives. The software vendor sent its implementation expert to facilitate the mapping of the workflow in the software, user-defined fields, work codes, and action codes.

The project team members took an initial look at the templated solution. They began asking questions of the vendor representative because they had not understood the value of getting control and stability of work management practices.

In days of old, prior to the software upgrade, a craftsman walked down the job, figured out what was needed, went to the stores counter and asked for the parts to fix the problem. The project team couldn’t actually see any benefit of division of responsibilities in using a planner/scheduler. Budget cuts had eliminated planner/scheduler positions three years earlier because they did not really seem to do much good; of course, they never properly implemented the role. It was believed that craftsmen could do the job plan, get the parts and so forth without bothering anyone else. Besides, they would fully understand the work that had to be done. They spent a lot of time walking to and from the job site, looking up parts, walking to and from stores two or three times, and being disrupted by other priority jobs.

Sometimes the parts weren’t available, or operations wouldn’t allow maintenance when the craftsmen asked to do the work. But it was OK because the backlog of work just meant there was “job security” to the project team members. In fact, there was so much work in the backlog that production availability was frequently impacted.

The vendor representative saw this scenario a number of times. He knew that carrying over reactive work management practices from the old software tool to the new

THERE WAS SO MUCH WORK IN THE BACKLOG THAT PRODUCTION AVAILABILITY WAS FREQUENTLY IMPACTED.

software tool would not significantly improve labor efficiency, production availability, or profitability. But he was a vendor, and the customer was telling him what the customer wanted. As far as the project team was concerned, profitability was someone else’s problem; job security was more important to them. After all, corporate was pushing this project; it wasn’t the project team’s idea.

The project team didn’t change how planning and scheduling were done. Craftsmen were presumed to be much better at self-planning; the craftsmen continued planning their own jobs, and they took advantage of mobile computing. But the craftsmen kept getting interrupted by other emergency or urgent work. The amount of work being accomplished didn’t change much. Six months after the software was implemented, labor efficiency and availability were still lower than anticipated. Profitability was down millions of dollars because of the costs of software license purchasing, software vendor consulting support, and labor hours consumed by the project teams at each plant.

The lesson: Plants miss the opportunity to reduce inefficient practices. To avoid this common scenario use a third party non-vendor to facilitate the process re-design and to stay committed to control and stability. Align the software to effective work management practices. Software is just an inanimate tool; it is not smart or stupid. ☺

Tom Moriarty, P.E., CMRP, is president of Alidade MER. Contact him at tjmpe@alidade-mer.com and (321) 773-3356.