

The Trap of Procedural Inertia

CRAIG SCHROTTER, CPIM

SENIOR APPLICATIONS ENGINEER

PROFITKEY INTERNATIONAL

Sponsored by



If you mention inertia to a physicist, they'll tell you it's the tendency of mass to resist acceleration. It's the reason a loaded shopping cart is hard to get moving, and then even harder to stop, let alone turn. Those of us who reside in the world of ERP implementation, however, know inertia can apply to a host of things beyond loaded shopping carts, and it can make those things just as hard to start, stop or change.

We see inertia in the everyday world of business activity. Regardless of the logic compelling organizations to abandon cumbersome, outmoded methods, they still hold on, white-knuckled, to those familiar but flawed procedures. Let's be honest, who among us hasn't fought the urge to throttle the next glassy-eyed, stoop-shouldered mouth-breather who absently shrugs and mutters, "But that's the way we've *always* done it?"

That kind of inertia, however, is nothing new. Countless articles and books have been written on the problem. Everyone knows about it, and everyone has developed their own techniques for dealing with it. So this paper won't waste the readers' time on yet another treatment of this same subject.

Instead, we will look at a far more insidious problem – one that can be much harder to address: The trap of *procedural inertia*.

Procedural inertia is different from what we see every day. It has an almost incestuous nature and follows a circular logic that makes it difficult if not impossible to recognize from within the organization. It spawns procedures around procedures and infests the organization like tendrils of some Chernobyl-born strangling vine. Most of the time it can only be seen from a vantage point outside the system, either by someone new to the company or by someone entirely outside the organization.

All organizations create procedures. These procedures are intended to ensure a consistent flow of work and information. They allow for quick and easy retrieval of data and materials. They promote consistent, satisfactory and well-documented quality. They let members of the organization know what's expected of them and what they can expect in return.

Procedures evolve from the needs of the organization: If ITEM-001 needs to wind up on SHELF-22 when it's finished, then a procedure must be developed to be sure the desired activity occurs consistently and without error.

Usually, such procedures are written to work in conjunction with a company's ERP system. In such a case, the procedure would no longer simply say, "When ITEM-001 is complete; put it on SHELF-22." Now the procedure must refer to the feature in the ERP system's inventory management module used to process receipts from the shop floor; describing in detail each field entry and mouse-click needed to tell the system that a quantity of ITEM-001 has been received from a specific work order and stored in SHELF-22.

But, what would happen if the people writing this procedure had an incomplete understanding of the ERP system? What if they didn't know there was a feature in the inventory management module for receiving materials from work orders into specific inventory locations? In such a case, the resulting procedure would likely call for an inventory transaction to be made that simply adjusts the respective quantity of ITEM-001 into SHELF-22.

So now, the inventory quantity of ITEM-001 would be correct, but the work order that produced ITEM-001 would be ineligible to close for lack of a complete production receipt. Over time, thousands of work orders would build up, each with closing exceptions.

To overcome the growing volume of un-closable work orders, a new procedure would be written calling for each work order to be manually closed once its production is received into stores. But without closing work orders through the process expected by the ERP system, the costs of ITEM-001 would not be properly updated. As a result, the inventory quantity of ITEM-001 may be right, but its value would eventually be meaningless.

To address what the organization now sees as a clear flaw in the ERP system's design, a new procedure would be written to calculate costs outside the ERP system. The procedure would likely involve a spreadsheet that would be manually updated upon the receipt of each work order for ITEM-001. Someone in Accounting would need to determine the costs of material, labor and burden and then input these values into the spreadsheet.

If ITEM-001 were the only item involved, maintaining such a spreadsheet might not be too overwhelming. But in reality, there are probably hundreds or even thousands of such items. So now, the spreadsheet probably becomes several spreadsheets and each spreadsheet probably has many subordinate worksheets. And now the staff required to maintain this information has grown from one individual working an hour or so a week, to at least one full-time employee.

And it gets better: What if ITEM-001 and its siblings are the top levels of multi-level structures? Now, in order to properly update the costs of these items, the costs of their subordinate items must be calculated first. But the subordinate items' costs cannot be calculated until *their* subordinate items' costs are calculated. And so on, and so on.

Ultimately, an entire department is required to research and correct erroneous vendor invoices and PO prices, missing inventory transactions and incorrect labor transactions; all of which must be done before these pieces of data can be entered into the costing spreadsheets. And of

course, a suite of reports must be written and maintained to extract, organize and present data from the spreadsheets in a meaningful way.

Remember, this all results from someone not understanding how to receive production from a work order into stores. But as far as the organization is concerned, this all arises out of a poorly designed ERP system.

So now a new materials manager arrives on the scene and can't believe an ERP system could be so badly designed as to require all this extra effort, so they contact the ERP vendor, who sends a consultant. The consultant recognizes the problem within the first few minutes, and explains the proper methodology. But a whole culture has grown up around this procedure. There are dozens of reports pointing at these spreadsheets. These reports are central to the functioning of Purchasing, Planning and Sales. Dismantling this structure and relying on what the company sees as a poorly designed ERP system is seen as madness.

That's procedural inertia.

And this example is only one of dozens. Here's another:

An organization's work centers are incorrectly configured and their routings are incomplete. Moreover, their labor reporting is grossly inaccurate. Not surprisingly, their repeated attempts at finite scheduling are horrific. So, they abandon their use of the ERP system's scheduling features because the poorly-designed ERP system doesn't work for their type of business (sarcasm intended) and they adopt a system of spreadsheets instead.

The spreadsheets must be manually updated by entering new orders, deleting completed and canceled orders, modifying orders with changes in dates or quantity and color coding everything. This spreadsheets become the central planning documents for the entire company.

Purchasing uses them to determine what orders' materials must be bought and when; Scheduling uses them to determine planned overtime and to establish a dispatch for the shop and Sales uses them for calculating backlog and forecasting.

A host of reports and related sub-procedures grow up around these spreadsheets to the point where, even when offered the opportunity to undertake effective finite scheduling, the company decides to retain its current methodology.

That's procedural inertia.

Here's another one:

Purchasing needs a report telling them what items' on hand quantities are insufficient to meet those items' requirements. They are willing to pay an ERP consultant to write such a report. Their current procedures involve the use of a spreadsheet, but it's too difficult to maintain. When asked by the ERP consultant why they don't simply use MRP, a long exchange ensues about MRP not working correctly. Ultimately, the root cause turns out to be an inaccurate inventory. But rather than change inventory procedures, the company chooses to proceed with their new custom-written report, notwithstanding the fact that it will be based on an inaccurate inventory.

That's procedural inertia.

The upshot from these issues – this procedural inertia – is not simply extra work, inaccuracy, inefficiency and a failure to fully exploit an ERP system that costs a significant fraction of a million dollars. By clinging to these procedures the company is effectively placing a cap on its ability to grow.

With a properly implemented ERP system, the limit to the organization's growth is determined by its capacity and its access to materials. So as business increases, the organization can grow to accommodate that business as long as these two components can be increased as needed.

But within the trap of procedural inertia, an artificial limit is placed on growth. As business increases, the organization cannot respond through its ERP system, since the ERP system is not the vehicle by which the organization moves information, materials and work – the procedures are the vehicle. The ERP system is simply an illusion.

And unlike an ERP system, which can accommodate and process increased input to an almost unlimited level, the manual procedures quickly become saturated. No matter how many people the organization employs to work the procedures, the system has an inherently limited throughput, just like a bottleneck on a shop floor.

In most cases, once an ERP consultant has been called to the scene, the procedures are already approaching their saturation point. But it's only just now that things are starting to slide: Materials remain unordered or arrive late; subordinate items remain unproduced or are behind schedule; scrap is up; quality is suffering and of course, deliveries are slipping.

Until this time, problems have been hidden from management because personnel have been working harder and expending extra time to ensure work gets through the process. These personnel are almost universally staff rather than line; so no records exist to indicate the extent of the extra time and work. In other words, no signals reach management. Now, however, with the saturation point approaching, the extra effort is ineffective, and problems start to become visible, hence management's call to the ERP consultant.

The manual procedures are what constrain the growth of the organization, and that constraint usually begins to operate long before those imposed by capacity and materials. Moreover, unlike a shop floor bottleneck that becomes visible almost immediately, the constraint imposed by the manual procedures gives no warning until it is suddenly the organization's greatest problem.

One way to deal with the effects of procedural inertia is to first acknowledge its existence. Then, look for telltale signs. Try to hear employees' complaints about tasks taking longer than they took in the past. Look for signs of personnel working through lunch, arriving early, staying an extra half hour after the end of their normal work days or coming in on days off. Look for stacks of paper where there used to be few or none. Look for delayed secondary tasks like reports that are late, incorrect or never submitted at all. Look for changes in people's demeanor: Are they more harried; less prone to chat or have they lost their senses of humor?

These are good indicators that your organization may be about to reach its procedural saturation point. And you may just pick up on them in time to take effective action. On the other hand, the horse may have already left the barn.

The right answer is to break the back of the convoluted system of extraneous procedures before it establishes a death grip. Ask people why they do things. Ask them how they became aware of their next task and to whom they pass work when they're finished. Ask them to explain their spreadsheets and reports— not just what they mean, but how each is used. Ask why things go where they go.

Understand how your ERP system works, what inputs it expects and what features it offers and how they operate. With this information, challenge your employees to defend their procedures

and insist those which are unnecessary be abandoned, no matter how great the wailing and gnashing of teeth.

Another excellent option is to have your ERP vendor send a consultant to do an in-depth work flow evaluation to determine what features are being under-used and where procedures can be streamlined.

Finally, throttle anyone who says, “But that’s the way we’ve *always* done it.”

About ProfitKey International

For over 30 years, ProfitKey International has been a leader in providing Manufacturing Planning, Manufacturing Operations, Financial Management software and Customer Service. Today, that tradition continues with new technology, new innovations, and new services. ProfitKey directly implements and supports Rapid Response Manufacturing[®] Software from seven offices nationwide. Over 1,200 customers with over 21,000 users have implemented ProfitKey software in the United States, Canada, Europe and the Far East.

PROFITKEY INTERNATIONAL

800-331-2754

WWW.PROFITKEY.COM

BLOG.PROFITKEY.COM

@PROFITKEY ON TWITTER