

AFTER THE MILLENNIUM BUG HAS BEEN EXTERMINATED

Mike Thacker, MIOM, SM Thacker & Associates

The replacement of an old system with a new system has its own special problems. When you originally implemented your first stock control or MRP system, the original process was manual and was replaced with a computer system. It had limited connections to other parts of the business, such as accounts or design. Nowadays you are replacing a computer-based process with a computer-based process - in theory that should be easier. Then why are so many companies getting into difficulties?

The race to beat the Millennium Bug has led to pressures on implementation deadlines and shortcuts not all of which were feasible or desirable. I have now been involved in a number of situations where not only have the original lessons been forgotten but also new lessons have been hard learned. Although this is not intended to be a comprehensive checklist, here are some of the most common issues I have encountered:

THE FORGOTTEN LESSONS

Project Management

- 1 The project needs an executive champion. It is *not* an IT project.
- 2 Manage the project with proper ownership of tasks, planning and control.
- 3 Not all data is dynamic but some is. In a staged implementation process this needs to be taken into account.
- 4 Parallel running the new and old system with dynamic data is extremely difficult.
- 5 Parallel running reveals as many problems in the old system as in the new system.
- 6 Software problems are a useful excuse for project delays but they are rarely on the critical path of delivering business benefit.

Business Issues

- 7 Identify the data owners. Operational responsibilities for the data including executive ownership need to be defined and agreed.
- 8 Computer salesmen do not provide solutions. You do.
- 9 Software suppliers (and their agents) rarely make good inventory planning and control practitioners.
- 10 Work is redistributed as a result of implementation.
- 11 Design the process taking advantage of the software.
- 12 You need to change the way you work not change the software to do it the old way. The new system is not a carbon copy of the old system and you should assume that *you* will need to change, not perpetuate the old malpractices.
- 13 GIGO. Parameters need to be maintained, with procedures defined and agreed.
- 14 Set up measures for the data accuracy and control it.
- 15 Educate the people not just how to use the software but also in best practice techniques, so that they understand the implications of their actions.
- 16 Run a conference pilot to simulate the running of your factory, *before* you go live. This needs the creation of a test script, which deals with all the routine and abnormal things that you need the computer system to handle. The running of the script must involve the stakeholders.
- 17 Set up disaster recovery procedures. You need a way of operating the factory or warehouse for a short while if the system is down. This is not an IT contingency plan alone. It could be any risk to the operation and it does

not mean spare computers, it means for example, being able to despatch without a computer system.

- 18 You did not design your original business process. You implemented the software. In some cases you superimposed the new computer system on top of the old manual system, so you increased the workload and manning and you still could not keep the data correct.
- 19 Complex computer functions (particularly modelling) require skilled operators and continuity of skilled operators.
- 20 Skilled computer operators are not necessarily good production and inventory control practitioners.
- 21 Complex software has lots of bells and whistles that need to be 'configured'. Configuring wrongly has dire implications!
- 22 Comprehensiveness (complexity) hides problems, lays mantraps and requires management.
- 23 Factories and warehouses are uncertain places, not lending themselves naturally to mathematical techniques.

IT issues requiring business management attention

- 24 Software modifications shorten the life of the software.
- 25 There needs to be a method of running the reports and housekeeping and still provide an on-line service especially in the evenings and weekends.
- 26 All software has bugs (some more than others and some more serious than others).
- 27 Many software packages do not discriminate between read and write responsibilities. I will allow anyone to look at a Bill of Material but I will not allow anyone to update it!
- 28 In the event of a system problem there needs to be the facility to recreate the transactions from the last backup. The worst kind of problem is a faulty recovery.
- 29 Programme and configuration change management is a significant, necessary, but unpopular process where the IT guys are seen to be holding up the implementation because they want to do things such as testing, version control, configuration management, documentation etc.
- 30 You need a test system for programmers use to test programmes, new releases, bug fixes etc. with its own set of programmes and databases. You need a training system with identical programmes and parameters to the production system (except when retraining after changes) but separate databases for users to use which is set up to run the conference room pilot and can be re-run at will. There needs to be sufficient computer resources to routinely operate these three environments and a change management process to move from one stage to the next.
- 31 You need a totally secure computer production environment including prevention of unauthorised access (accidental or deliberate), virus protection, fire, theft, malicious actions of current or former employees, collusion in fraudulent activity etc.

THE NEW LESSONS

Project Management

- 32 The same rules above still apply.
- 33 Parallel running is not an option, when replacing an on-line system.

Business Issues

- 34 Your business has changed since you designed the original system. The original system was aimed at a

highly functional organisation where the words 'best practice' were not words on anyone's lips. The current business has a totally different product set with different customer and supplier profiles, different organisation, (hopefully process focused). The business priorities have certainly changed with simplification in some areas and sophistication (complexity) in others. You need to re-design your processes.

- 35 Also because the software is different you cannot map your old processes, data, parameters, code structures and transactions directly onto the new software. Quite often there is a completely different hierarchy of data with different pre-requisites. The processes need migration as well as the data, parameters and code structures. Then the new business transactions can be mapped onto the new software.
- 36 Your original software did not reflect subsequently revised inventory planning and control thinking. Using the old software as a checklist to buy new software misses the opportunity to exploit the new thinking using new functions.
- 37 Inventory planning and control has changed since you designed your original system, making some of the original software functionality provided, obsolete or inadequate. It should not be replicated in the new software.
- 38 Archive the old data. Do not migrate it. Now it is easy to migrate the data using computer programmes, but do you really need all those old routings for those obsolete products based on that plant which was sold 3 years ago?
- 39 Policies, procedures and facilities for archive and retrieval of data are essential.
- 40 The new software is different. Because it is a package it will satisfy a *different* but hopefully larger sub-set of the complete requirements (if you have done your software selection correctly). Do not assume all your old programme changes are still necessary, the new software may simply do it a different way.
- 41 The dirty data tolerated by the new system is different and the new system may be tolerant or intolerant to different data from the old system.
- 42 The well-established technical support infrastructure you used to enjoy and undervalue ("we never had these problems with the old system"), does not exist any more and has to be re-invented.
- 43 The people still need education as well as software training. This is a great opportunity to do that, but why did you wait until now? Ideally education should be part of the induction programme for new employees or part of the change management process for system changes. It can also act as a catalyst for change in a workshop style session. One operations director said to me that he aimed to repeat education at two-year intervals to simply remind people what they were supposed to be trying to achieve.
- 44 The conference room pilot is even more important and change-management using an updated conference room pilot scenario essential. It must also be used in testing configuration changes. The conference room pilot must also check the validity of the migration path and especially the migration programmes.

IT issues requiring business management attention

- 45 There is even more software around now from which to choose.
- 46 Some of it is even more dysfunctional, lacking in useful inventory planning and control functions and less efficient to use than the old software. One I was using recently contained all these faults.
- 47 It is more comprehensive (complex) than the old software.

48 The IT guys have to learn the new software as well as a new technical environment, at a time when implementation pressures preclude proper training.

49 Migration programmes cause more problems than they solve.

50 So-called technical advances are in fact limited, non-existent or retrograde from a user point of view. Examples of this are:

- 'forward recovery' from last night's backup (which in some cases requires re-inputting all the day's transactions),
- 'dynamic transaction back-out' which avoids database corruption in the event of a power cut, or transaction failure.

I am old enough to remember that these were *never* allowed to become a business problem because technical facilities existed to avoid the problem. Also:

- 'Client/server' and 'GUI' are fine until there is a problem.
- 'GUI' is not necessarily a good way of high speed transaction processing.

51 Integration is fine provided you can undo an accounting transaction that arose out of a faulty original transaction (for a faulty goods receipt for example).

52 Integration is not simply a technical issue it is also a training and communication issue, in that the person originating the transaction must understand its implications in other parts of the system.

53 Integration is an overused and abused word, often meaning that there is a loose connection between the applications, and that in some circumstances they can become out of step, or at least that for the duration of an on-line day they are not consistent until the overnight interface is run.

54 Reliability, availability and performance of computer systems are forgotten words, replaced by the words, "the system has crashed again", "we have to take the system down at lunch-time again", and "it always takes five minutes to log on" respectively. The worst one of all I encountered recently is "the IT department's door is locked they must have gone home". But perhaps I am showing my age again.

So you have felt the pain. It is still not too late to get some business benefit out of the new system. If you changed your software to exterminate the Millennium Bug alone, you are definitely missing an opportunity. Go back and check the above list and start work on it as soon as the dust has settled on the implementation. If you are in the fortunate position of not having been chasing the Millennium Bug you still have time to do it right first time.

About the author

Mike Thacker MIOM is an independent management consultant. He has completed over 150 projects selecting, designing, implementing and diagnosing problems in a wide range of ERP, MRP, JIT, Advanced Scheduling, Cellular Manufacturing, Supply Chain and other manufacturing control systems. Industries he has worked in include aerospace, automotive (first, second and third tier), food, fast moving consumer goods, electronics, telecommunications, capital goods, spares, repair and overhaul, pharmaceuticals, basic metals refining, smelting, casting, sintering, forging, chemicals, printing, and plastics. These include a number of blue chip companies and small to medium enterprises in the UK, USA, Continental Europe and India.

He has been involved in four university research projects in best practice manufacturing planning and control, and managed the "JIT/MRP" project, which integrated the two philosophies at Lucas Industries.

He has worked as Production Director, Logistics Manager and Senior Consultant. Before that he had responsibility for many MRP implementations and systems management in Lucas Industries and GKN.