Integration of Process Manufacturing Systems; Benefits and Pitfalls

Evan Garber

Section 1 – Integration environment
Section 2 – Methodology Types
Section 3 - Summary

Section 1 – Integration Environment

As you are researching process manufacturing software, you may notice that there are very few options to choose from. Process manufacturing is a relatively small market when compared to markets such as discrete manufacturing, retail and professional services; this is evidenced by the number and quality of software solutions for each of these industries. Discrete manufacturing, retail and professional services have dozens of software solutions; from entry level through Tier One. The competition for ERP market share in these industries is highly competitive. Process manufacturing, on the other hand only has a few software solutions making the industry only lightly competitive for ERP's. The process manufacturing industry has four key issues which keep ERP publishers away from the market:

1) A relatively small market compared to other industries.
2) It is a highly specialized industry, requiring specialized knowledge to develop adequate solutions; i.e. a high cost of entry into the market.
3) The requirements of the industry are such that discrete manufacturing software will not provide an adequate solution, i.e. tweaking an existing discrete system will not work.
4) Within process manufacturing there are many specialized vertical markets, so the software must be broad as well as deep.

With these four key issues of process manufacturing software development, the cost of entry for most ERP publishers is too high for an industry that is too small. However, ERP publishers cannot overlook that there is a vertical industry with a need. Therefore, publishers have added process manufacturing solutions to their ERP by relying on third party solution providers (third party) to integrate with their ERP. Most of the time, the third party has at least some industry experience with process manufacturing and at least some experience writing and integration systems. With this type of model for integration, a buyer of process manufacturing ERP systems needs to:

- Understand the level of experience the third party has with both process manufacturing and developing software.
- Be aware of the integration methodology used by the third party.
- Understand the benefits and pitfalls of the methodology chosen by the third party.

As we walk you though our analysis, understand that if you are evaluating a "mainstream" ERP from publishers such as Sage, Epicor, SAP or Microsoft the ERP publisher has exponentially more resources (financial and human) than a third party developer. Therefore, as a prudent buyer of software, you will need to be aware of the third party's ability to support your chosen software for the long-term. As we look at the methodology types, we will talk about how heavily your company is dependant on the third party. Dependence on the third party isn't necessarily a negative; since working with a system that "fits" is generally better than the alternative of running your business on Excel or a discrete system. Software evaluation should be a process of assessing and mitigating risk your company's risk, this paper is intended to help you identify areas of benefit and risk.

Section 2 – Methodology Types

There are three methodologies used by third party developers of process manufacturing software:

1) Buy and change the ERP source code and a develop process manufacturing add-on
2) Generic system with an interface to multiple ERP's
3) Built in the framework of one ERP

Type 1 - Buy and change the ERP source code and develop a process manufacturing add-on

This method was used more frequently in the late 1980's and early 1990's and is typically not used anymore; though there is still a system on the market that uses this methodology. With this methodology the third party purchases the source code of an ERP and changes the source code of the entire ERP (accounting and distribution) to facilitate their process manufacturing add-on.

Benefits:

- Control. The developer has complete control of how the ERP interacts with their process manufacturing add-on.

Pitfalls:

- Ability to upgrade. This method takes the customer out of the ERP developers upgrade path. As technology changes with the ERP, the third party system won't follow along – over time, users will be stuck with software that uses outdated technology.
- Dependence on third party. You are now relying on the 3rd party developer to maintain and enhance your **entire** system (accounting, distribution and manufacturing) instead of just the manufacturing. Do they specialize in manufacturing, accounting or distribution? Can they specialize in all three?
- Add-on distribution modules. Since they have changed the underlying code of the ERP, the burden will be on the third party to develop add-on's such as warehouse

management, customer/vendor web portals, EDI, etc. Can they/will they provide these applications for your growing company?

- Financial stability? Since the **entire** system relies on a 3rd party developer, a user must be comfortable with the third party's financial stability and ability to support the software and continue to evolve the software over the long run.

Since this method is sometimes confused with a Type 3, Built in the Framework, ask the third party the following questions:

- Who maintains the upgrade of the accounting and distribution modules? (if the answer is the third party, the software isn't a Type 3)
- Did you change the source code of the accounting or distribution modules? If so, would I be out of the upgrade path from the original publisher? (if the answer is "yes" to both, this isn't a Type 3).

Again, this method is not frequently used in the market today since, in the '80's and '90's, many companies found themselves with software that couldn't be supported because the third party was no longer in business.

Type 2 - Generic system with an interface to multiple ERP's

This is the most common method of process manufacturing third party solution used today. With this methodology, a third party develops a process manufacturing solution then interfaces their software to one or many ERP solutions.

When **interfacing** two systems, data from one system is *translated* and imported into another system, typically an unposted transaction file. The transactions then need to be posted in the ERP; therefore interfaced systems do not provide real-time transaction processing. In contrast, **integrated** systems read and write directly from the native data tables and provide real-time transaction processing.

Generally, Type 2 systems have the following benefits and pitfalls:

Benefits:

- Inventory Control. The ERP source code is not changed, so the user can keep remain the upgrade path with the ERP publisher. Therefore, many of the ERP modules will be on current technology and evolve with industry enhancements.
- Other ERP's. The third party will usually interface with many ERP's, so if you change ERP software, there is a chance that they can interface with your new system.

Pitfalls:

- Data flow. There is no real-time transaction processing, i.e. a closed production batch will require an inventory transaction posting in your ERP in order to update inventory balances and costs.

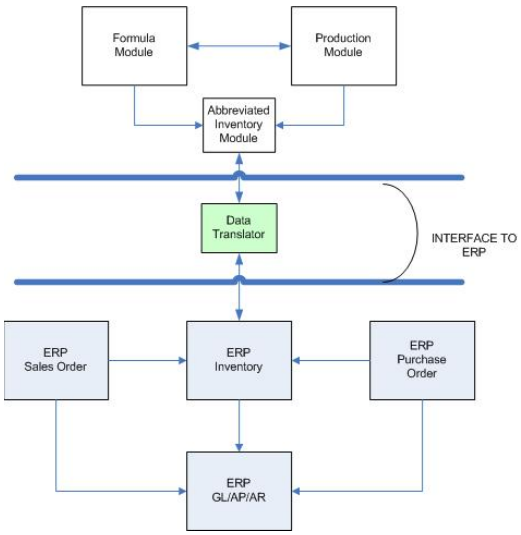
- Systems Maintenance. This methodology requires maintenance of two separate systems with two separate interfaces and two different technologies.
- Denormalized Data. Since the two systems save redundant information, data can easily become out of synch. For instance, both systems keep inventory SKU# and description. If the description changes in one system will the other system pick up the change?
- Third party dependence. You are relying on the 3rd party developer to maintain the interface to *all of the ERP systems* that they work with (especially yours). Most ERP systems have monthly or quarterly releases as well as major upgrades. If the third party interfaces with five ERP systems, each with an average of four releases per year, that is 20 interfaces that need to be managed per year; over 5 years there are 100 interfaces to manage. Can your 3rd party provider maintain integration to hundreds of versions of many ERP's?

There are two types of Interface Systems on the market today:

1) Third party manufacturing Modules Only, and
2) Third party manufacturing and Distribution modules.

Interface Method #1 – Third Party Manufacturing Modules Only

The first methodology commonly used by third parties to develop an interface to an ERP is the "Manufacturing Modules Only" approach. In this method, the third party develops only the modules required for manufacturing such as the formula, production and quality control modules. There is typically an abbreviated inventory module to hold basic inventory information from the ERP; a synchronization utility is used to gather necessary information from the ERP. See the diagram below for a data flow.



Benefits of Interface Method #1:

- This method is known as a "thin interface" since it doesn't replace any of the modules of the ERP. Typically that means you have a better chance of keeping up with current technologies and best practices on your accounting and distribution modules.
- You have a limited amount of exposure if the third party were to stop supporting the software since they only support your manufacturing processes.

Pitfalls of Interface Method #1

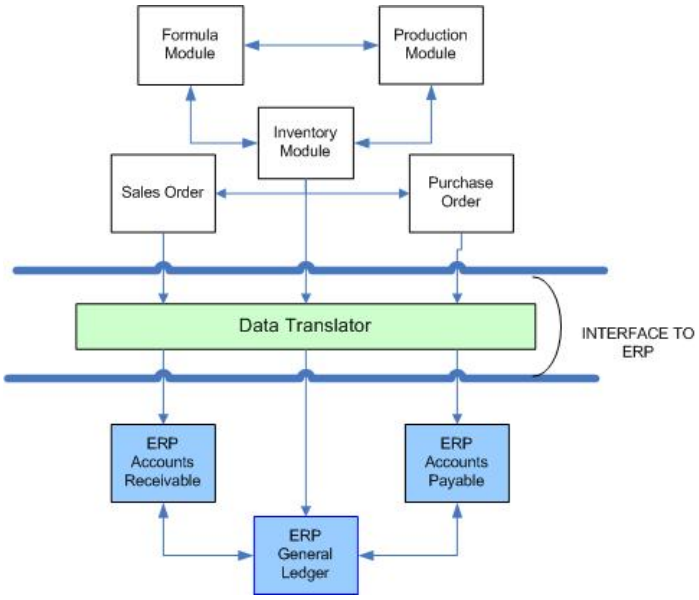
- Inventory costing is always an issue with this methodology. Since inventory transactions are produced by the third party and posted by the ERP, the third party software doesn't know what the actual costs of a production batch are; reconciling the two systems is nearly impossible.
- Limited reporting out of the box. Since the third party may integrate with multiple systems, they usually don't have many reports. The reason for this is simple; to gather information needed for a meaningful report typically requires information from multiple modules, i.e. production, inventory (costing) and sales order. This information would have to come from disparate systems.
- No real-time flow of data. In an interfaced environment such as this, a batch can be closed but the ERP inventory is not updated until a user posts the inventory transactions in the ERP.
- MRP/Planning is typically limited. The disconnect between the production and inventory/sales order/purchase order modules make it difficult, if not impossible to have a working MRP.
- Reversing transactions. Many times an inventory transaction needs to be reversed. With this method of development, reversing production transactions is generally not possible.

Interface Method #2 – Third Party Manufacturing and Distribution Modules

The second methodology commonly used by third parties to develop an interface to an ERP is the "Third Party Manufacturing and Distribution Modules" approach. In this method, the third party develops both the process manufacturing and distribution modules and interfaces to the general ledger, accounts receivable and accounts payable of the ERP. Typically with this architecture, the third party has to synchronize customers, vendors and general ledger information from the ERP to their software on a regular basis.

The third party software handles inventory costing of transactions but must interface with accounts receivable for sales order shipments and returns, accounts payable for

purchasing receipts and returns and general ledger for production and inventory transactions. See the diagram below.



Benefits of Interface Method #2:

- Inventory Costing. The most significant reason for this method as opposed to Interface Method #1 is that the third party can control the inventory costing of transactions. Realizing that inventory costing is the most complex task in any ERP, if the third party does it correctly, your system will be fine, if the inventory module isn't written correctly, your company will be in real trouble. Remember, third party developers have fewer resources than ERP developers.
- Reporting. The reporting of this method can be better than Interface Method #1 because the distribution and manufacturing modules are related.
- MRP is possible. Since distribution and manufacturing are from the third party, an MRP solution is possible.

Pitfalls of Interface Method #2

- Operational dependence. Your entire operations rely on the third party instead of the ERP publisher. You must feel comfortable that the third party is financially stable enough to support their product in the long-term.
- Experts in all operations? You are relying on the third party to be experts at both manufacturing and distribution and to include all of the features that are necessary for your business.
- Add-on distribution modules. In a distribution environment many companies require add-on's such as warehouse management, customer/vendor web

portals, EDI, etc. Even if these add-ons aren't important to you right now, they may be in the future. Determine if the third party has the ability to provide these modules to you.

- Denormalized Data. Since the two systems save redundant information, will all of the changes made in one system synchronize with data in the other?
- Current technologies. As technology changes will your third party have the ability to keep their product up on current technologies in the long-term.

Type 3 - Built in the framework of one ERP

If the ERP publisher were to write a process manufacturing system, this is the method that they would use. This method is very difficult to find in the process manufacturing arena especially in the mid-market; but they do exist. Third party solutions that have been developed in the framework of the ERP have complete or seamless integration to the rest of the ERP. In addition, the user interface is similar to that of your ERP. Essentially this method includes one look and feel, one technology, one system. If the framework has tools for customization or reporting, the third party solution will typically be able to utilize those tools as well.

Benefits

- Full ERP solution. You will be able to use all of the modules of the ERP. The process manufacturing modules will just "bolt" on.
- Inventory costing. Inventory transactions will be actual costing in real-time.
- Maintaining current technologies. This method will allow you to stay current with technologies on the entire system not just pieces.
- Integrated vs. Interfaced. Since modules are integrated (vs. interfaced) there is a real-time exchange of data throughout the entire system.
- ERP Tools. If the ERP provides productivity tools, the third party will be able to leverage those tools, i.e. screen customization, reporting, etc.
- Add-on modules for distribution. Since this methodology uses the publishers distribution modules, there will probably be choices of add-on modules for warehouse management, customer/vendor web portals, MRP, etc.
- Integration maintenance. The third party is focused on integrating with only one ERP and will only be required to maintain a hand full of versions over the years.

Pitfalls:

- Third party dependence. The developer will need to maintain the integration to new versions of the ERP. However, this is typically mitigated because the ERP publisher could take over the development of the third party software if the third party is no longer able to, since the application is written in the ERP framework.

Final Notes:

As we have discussed, third party developers have fewer resources than ERP publishers. Because of the lack of resources third parties sometimes skimp on the little things that make a big difference when running a manufacturing system, such as:

- An instruction manual – either paper or electronic
- Online help which is screen sensitive.
- Quality control methodologies for a new release and patches.
- A support staff. If you pay annual support and maintenance, is there someone who will be available to get back with you in a timely manner?
- Web support for logging incidents and a knowledge base for self-searches.

If the third party doesn't have some of these items you may question their ability to support you for the long-term.

Section 3 – Summary

As you search for process manufacturing systems, you may notice that there are very few options to choose from. This is due to the high cost of entry for a relatively small market sector. Because of this, ERP publishers typically team up with third party developers to provide a process manufacturing solution.

The process of researching software should be a risk-assessment/mitigation exercise for you and your company. When your process manufacturing options are "bolt-on's" from a third party, you need to understand the type of integration/interface that the solution provider uses and the benefits and pitfalls of each method. By being fully aware of your options and risks, you will have a greater chance of success with your new process manufacturing solution.