

***FLEXIBLE PRODUCTION CONTROL -
A FRAMEWORK TO INTEGRATE ERP WITH MANUFACTURING EXECUTION
SYSTEMS***

Jan Philipp Louis , Paul Alpar

Philipps-University

Abstract

Manufacturers are under pressure to improve their shop floor processes for more cost effectiveness. At the same time they are required to adapt their products to fast changing customer demands and regulatory constraints. This can be achieved through IT-support but the flexibility of human decision-making should not be decreased. To meet these requirements the transparency of shop floor processes needs to be increased, traceability throughout the supply chain should be assured or improved – even beyond organizational boundaries - while labour cost should be decreased at the same time.

ERP-suites are not up to the task of providing a suitable IT-support for shop floor processes due to certain limitations although they are the informational backbone of many IT-infrastructures. Manufacturing execution systems (MES) are information systems which are specifically designed for facilitating shop floor processes. This paper proposes a framework for the integration of an ERP-suite with a MES to provide an up to date support for shop floor processes. Based on this framework, a systematic approach is shown for designing the integration and implementing a MES.

Keywords: *manufacturing execution system, shop floor processes, total cost of ownership.*

1 INTRODUCTION

The flexibility of manufacturing firms is determined by the adaptability of their own shop floor processes. Short time-to-market, low costs, fast adaptability to customers' needs and compliance with regulatory constraints are fundamental requirements for entrepreneurial success nowadays. To meet these requirements shop floor processes need to become more flexible, reconfigurable and adept. This is again closely related to the provision of an adaptable information systems support for the shop floor processes.

The information system support for shop floor processes, like most business processes, can be provided not only by individually developed programs, but meanwhile by standard software. In case of support by standard software, two major approaches for shop floor support have emerged (Schmidt, 2004). The first approach is to use an enterprise resource planning (ERP)-system which nowadays represents the core of the IT-architecture in many enterprises. There is a common trend to extend the functionality of these systems by customizing them or by adding functions to support shop floor process. In contrast to

this approach, the application of manufacturing execution systems (MES) has been established; a relatively new class of information systems that are specifically designed for supporting shop floor processes. They are used as platforms to integrate the IT-support of shop floor processes with the information system architecture of an enterprise.

This paper substantiates why the first approach – the only ERP-based solution - is constitutionally not sufficient to meet the actual requirements of manufacturing enterprises. The critical part of the second approach is the integration of a MES with the existing IT-infrastructure., In particular, a tight integration with the enterprise-wide ERP-systems is necessary. Therefore, an integration framework is proposed for enterprise resource and manufacturing execution systems to enable the enterprises to meet the market requirements and to adapt fast and at low costs to emerging trends. Based on the presented framework a systematic approach with four major steps is introduced for integrating enterprise resource planning with manufacturing execution systems.

2 LIMITATIONS OF ERP-BASED SHOP FLOOR SUPPORT

ERP-systems can be seen as the informational backbone of the IT-infrastructure within an enterprise. Their main task is to provide an integrated database and functional modules across departmental boundaries and throughout a global enterprise (Bendeich & Strohmeier, 2004). An ERP-suite consists of a big collection of enterprise-wide functions. Such systems already provide functions for shop floor processes support, but their main focus was traditionally on the planning side of the production processes. In particular, this was executed by providing scheduling algorithms like MRP-II or newer ones to perform production planning and a requirement-based purchasing of supplies (Bendeich, 2005). One of the main problems of ERP-system production modules is their lack of integration with real-time production data generated at the shop floor (Reilly, 2005; Wannewetsch & Nicolai, 2004). This is why many of the resulting production plans cannot give guidance when quick reaction to changing demands or emergent deviations is needed. To achieve a better level of support for the shop floor processes, ERP-suppliers started to offer new functions or to integrate whole software products. But despite these additional functions, some more fundamental differences between the requirements of the shop floor processes and the characteristics of an ERP-suite remain.

Information systems which support critical shop floor processes have to meet very strict robustness requirements. While a down-time of two hours may be accepted for an ERP-system, in most industries this is not an acceptable down-time for shop floor systems, because this could lead to a total halt of the production itself and, with that, to a total loss of all currently used material (Loos, 1997). A great gap can occur between these requirements and the capabilities of a regular ERP infrastructure.

In a production system, there can be great process flexibility for a given production order. In particular, within the piece-oriented production, a large variety of production routings through one production department can be observed. Besides resource substitution and changes in production sequence, there can also be a substitution of whole sub processes with alternative ones. To provide an adequate support to the shop floor processes an ERP-system should have the ability to deal with the given production flexibility. As Trebilcock already stated (2004) this becomes a problem for ERP-systems because they

are originally designed for well-structured and mostly linear processes (like financial accounting).

The real-time data, which result especially from the use of automation systems, lead to enormous data amounts and require a very short response time from the receiving system. The information system must be able to deal with *substantial* amounts of data without any scheduling problems in a smart way. This is a task ERP-systems cannot accomplish (Beuthner & Fritsch, 2004).

Another aspect is the required level of detail. For a suitable shop floor support a very detailed modelling of the real production processes is needed. The detail level and the type of data depend on the shop floor processes. For manual production processes, human readable instructions are required, but for the automation system control recipes with a bigger number of computer readable parameters have to be provided and transferred to the corresponding systems. In any case the needed detail level is much higher than any standard ERP-system is able to provide because for the major tasks of an ERP-suite (e.g., accounting) these detailed and technical data are not relevant and in consequence not stored (Trebilcock, 2004). An issue related to the level of detail is the strict accountability of executed processes. For processes in some industries, e.g., pharmaceuticals, all information about machines, materials, and people involved in a production batch must be recorded. Much of this information is of no relevance to other ERP-functions and therefore not recorded there. This information, however, is mandated by governmental authorities like the Food and Drug Administration (FDA). It can be seen that these traceability constraints are becoming more important in more and more industries.

To summarize, the following fundamental differences exist between the requirements for the support of shop floor processes and the capabilities of ERP-systems:

- available functionality
- robustness
- flexibility
- real-time ability
- detail level and, related to it, traceability

These gaps between requirements and capabilities are hard to overcome by continuous evolution. Due to these facts some leading ERP-suppliers started recent initiatives to develop and provide standardized electronic interfaces to enable the integration and collaboration between ERP- and manufacturing execution systems (Kletti, 2005). As an example, the market leader SAP started the “adaptive manufacturing initiative” (SAP, 2004). Thus even ERP-suppliers try to overcome the above limitations by integrating MES. The question now arises as to how to design this integration and particularly how to distribute the functions between the two systems. In the next section, an integration framework is introduced from which a systematic approach will be derived as to how to integrate a MES for shop floor processes in a way, that the actual market requirements can be met.

3 MES-BASED SHOP FLOOR SUPPORT

To date there has been no commonly accepted definition of manufacturing execution systems as a class of information system, perhaps because of a lack of scientific research on this topic combined with exhaustive use as a keyword for many purposes (Mertens, 2005). Aside from several industry, company or country specific definitions there are two main international attempts to define MES. The first one has been developed by a group called Manufacturing Enterprise Solutions Association (MESA, 2000) and the other stems from the Industry, Systems, and Automation Society (ISA).

ISA implicitly defines within its S95 standard a MES by identifying four main categories of a MES-layer: Production, Inventory, Quality, and Maintenance Operations Management. Each of the categories contains eight function groups. This ISA definition is very rigid because dependencies between shop floor processes and characteristics of a specific MES-layer cannot be accommodated (ISA, 2000). Nevertheless, the four identified categories and their function groups are commonly accepted and can be used as a base to define the functional scope of the MES-layer. An XML-based standard interface language exists (called B2MML) which is already implemented in the SAP NetWeaver product and is used to implement interfaces between MES and ERP-layer (WBF, 2005).

We define a MES by the requirements it should accomplish: A MES is a computer information system, which supports

- production execution and control,
- supply and usage of real-time data of the whole production process and
- a rapid reaction to occurring deviations and
- the integration of automation systems and manual production processes with an ERP-system.

With this relatively abstract definition it could be possible to constitute a common industry independent understanding of a MES. For the further work here a more detailed definition is needed. So with adopting the S95 standard the functional extent of a MES can be described in a two-tiered hierarchy in four main functional modules – production, maintenance, quality and inventory. The production functions support all activities which are directly connected with the physical shop floor processes and thus with the creation of the actual products. Within the maintenance module the task of planning and performing maintenance at the shop floor are supported. The assurance of defined production quality and the generation of quality reports are the main tasks of the quality module. The inventory functions facilitate the transport and inventory activities within a production area.

To detail the functional modules a generic model is applied (see Figure 1). The functional modules are broken down into eight function groups with their dependencies. In particular, in part three of the S95 standard the interaction within the MES-layer is examined (ISA, 2005a). Although the detailed definition of the eight function groups within the generic model is not within scope of this paper (this is done in (ISA, 2005a)) that level of detail will be applied for the MES layer in the framework.



Figure 1 Generic model of the functional modules of a MES

In contrast to ERP-systems MES were originally developed for supporting shop floor processes. Thus the shown fundamental gaps of ERP-systems to the requirements do not exist or at least not to that extent. MES needs to be tightly adapted to the regarded production process. This is mostly done by customizing or adapting the software and modelling shop floor processes. The amount of software adaptations needed is significantly lower than those for an ERP-based solution, simply because there are no fundamental gaps to be solved.

Two important points about the definition have to be considered. First of all the above presented functional extent is very comprehensive and so it can not be seen as a constitutive characteristic for a MES to support almost every function group. The second is that also systems which are definitely not a MES like ERP-systems can support some of the function groups presented here. Thus the proposed detailed specification of a MES by its functional extent should be here further operationalized.

3.1 Integration framework for ERP and MES

Figure 2 shows the proposed integration framework for enterprise resource planning and manufacturing execution systems. The ERP-layer at the top of the pyramid provides global, enterprise wide function sets and an integrated database. The detail level is rather low and the planning horizon ranges in months or quarters. On the bottom of the framework, the automation system layer works generally in real-time and at a very high detail level. The boundary of the layer is determined by the connected production lines or machines and because of that this layer is very specific to each production department. In this framework, a MES-layer is located between the ERP-Layer and the automation system layer (see also Figure 2). It includes all functions which are necessary to provide a comprehensive IT-support for shop floor processes. Thus in fact the MES-layer includes the four functional modules presented above with their immanent function groups. This generic MES-layer within the integration framework gives the possibility to distinguish between original MES functions and case-specific functional support of given information systems.

The two traditional layers – ERP- and automation system layer - are mainly loosely coupled because of great differences between them and there is no or no tight interaction between them. This integration gap has also been identified in the context of the computer integrated manufacturing (CIM) concept as the missing link between the administrative and the technical branch of the Y-CIM model (Adam, 2001; Scheer, 1998)., Therefore a successful implementation of a MES as an integration layer to tightly

connect the shop floor processes into the enterprise wide ERP-system could lead to a revitalization of the CIM concept in future.

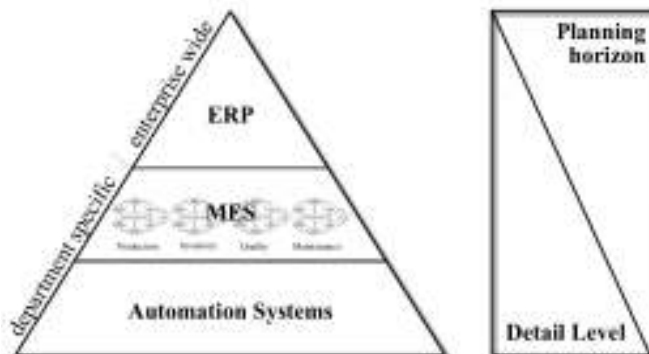


Figure 2 Integration framework

The above defined MES-layer located between the ERP and the automation system layer has interfaces to both of them. It receives data from the automation system layer either in real time directly from the automation systems or in an aggregated form, e.g., via a history database. It sends more aggregated data to the ERP-layer. The MES-layer accomplishes the integration of the automation systems of a production department with the enterprise-wide ERP-layer. The MES-layer has to implement two interfaces towards this goal. The first one is the interface to the automation system layer which is highly standardized and mostly determined by machine or supplier specific automation protocols (e.g., Siemens). The development of this interface is not further discussed here because there are only a few degrees of freedom and it represents a relatively clear, mostly technical implementation task. In contrast, the interface between the MES- and the ERP-layer has to be designed conceptually with a high level of design creativity. This issue will be examined in the next section.

Due to immediate proximity of the MES to the shop floor processes there are strong dependencies between the MES-layer and the characteristics of these processes. There is a need to customize the MES-layer according to the specific production department with its automation systems and shop floor processes. Thus both MES and automation systems layer are strongly characterized by department specifics and in terms of the MES-layer at least different priorities of modules and function groups have to be estimated for different production departments. Contrary to that the ERP-layer is an enterprise-wide layer and acts as the backbone of the IT-infrastructure. This again turns out the integration character of the MES layer to hook each production department to the enterprise wide information systems.

3.2 Approach for implementing MES

Based on the established integration framework, a systematic approach for implementing a MES is proposed. For this approach four major steps have been identified. These steps are

- deciding which function groups are necessary (determined by the shop floor processes),

- identifying the current IS support for each function group,
- choosing the right system for each function group and
- designing interfaces (especially between ERP and MES) based on step three.

In the implementation process, the integration framework is customized step by step to fit the needs of a concrete production department. The underlying assumption is that a suitable MES layer definition is only possible based on requirements of a concrete production respective of requirements for IT-support of shop floor processes.

3.2.1 Identification of necessary function groups

As discussed above, the requirements for specific function groups depend strongly on the characteristics of the shop floor processes, e. g., the requirements for a continuous, automated mass production differ fundamentally from the requirements of individual discrete production. The consequence is that for each production department an individual analysis must be performed to examine which function groups need to be supported. The examination of shop floor processes can be conducted through the study of process documentations or interviews with the affected users. Also, a first approximation of cost and benefits should be created. As a result, a specific MES-layer is defined, in which function groups are identified for which support by an information system is needed.

3.2.2 Identification of current support by IS

In this step, the function groups which are already supported by IS are analyzed. The IS could be the ERP-system, individually developed programs, or even another software package. Thus, steps one and two represent the analysis of the status quo. They form the basis for the next two steps.

3.2.3 Design of the future function distribution between ERP-system and MES

In the most simple case, a firm would require IT-support for all function groups and they would all be only supported by the MES. Further thoughts about a function distribution would be obsolete. Using the integration framework as introduced in section 3.1 would be sufficient. But as argued before, different requirement specifications can be expected for different production departments and so for each one requires a detailed analysis is needed to decide which function group to support by which information system. We focus here on an enterprise resource planning system and a manufacturing execution system as most likely candidates in this context.

Usually, only a few function distribution scenarios will be feasible so it is not necessary to consider all theoretically possible combinations, i.e., function distributions. Figure 3 shows a hypothetical scenario of a function distribution of the MES-layer designed in this way.

MES-layer

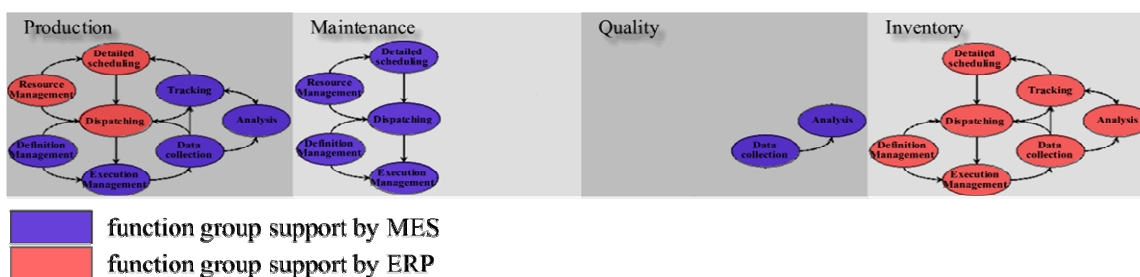


Figure 3 Scenario of function distribution of the MES-layer

The total cost of ownership (TCO) model of the Gartner Group distinguishes between direct and indirect costs (Wild & Herges, 2000). The relevant direct and indirect costs that need to be assessed are shown in table 1. These costs must be considered for every software that is considered within a scenario. The total cost of ownership for a scenario is thus the sum of individual software costs. The cost items are not independent of each other. Business specific software adjustments of standard modules not only cause costs by themselves but they also usually lead to higher downtime costs caused by software failures because such software is more failure prone than the well tested standard software. Furthermore, extensive adjustments can lead to a loss of compatibility with new releases of the underlying standard software package. If new releases are adopted, which sometimes simply happens because the supplier stops maintaining old releases, adjustments need to be programmed again.

A consideration of TCO often helps to realize that adjustments to standard software packages cost more in the long run than alternative solutions. However, that does not mean that they do not make sense in all cases. The weak point of the TCO model is that it considers only costs but not the benefits (Gibbons Paul, 1997). If the adjustments increase the adaptability of production process significantly beyond the exclusive use of standard packages, this may be profitable despite higher TCO. The benefits may include lowering cycle time or capabilities to accept unplanned customer demands.

Direct costs	Software	Licenses
		Adjustments
		Interface design
		Support treatment
	Project	Project realization
User training		
Indirect costs	Downtime	Downtime caused by HW failure
		Downtime caused by SW failure
	Maintenance	Update costs
		Upgrade costs
	Support	User Support

Table 1 Direct and indirect costs items for software products

The result of this step is a decision for one scenario which should be based on TCO and its related benefits. With that a case-specific design of the MES-layer it is shown which functions will be carried out by which package, whether the package will be customized for some functions, and which functions will be individually developed from scratch. This design also clarifies the border line between the ERP- and the MES-layer. This specific framework is the base for the next step in which the interaction between the ERP-system and the MES needs to be specified based on the involved function groups.

3.2.4 *Interface design*

The S95 standard defines a detailed data model for the interaction between a business and a manufacturing information system. The current version of the interface definition is rigid in the sense that it only works for a complete MES-layer support by one information system. When the MES-layer is adapted to a different situation, there is a need to define specific interfaces. The communication between each of the function groups will be defined in the forthcoming part four of the S95 standard so that this standard could evolve to a complete framework for interface definition (ISA, 2005b).

The chosen function distribution and the selected software packages form a clear specification for the examination of data to be exchanged. Technically there are two possibilities to transfer data between an ERP-system and a MES: by manual intervention or through an automated program. The manual transfer is only suitable for a small amount of data. For large amounts of data and rapidly changing data an automated transfer via electronic interfaces is more suitable. In the concrete definition of an electronic interface between an ERP-system and a MES the detail level of the transferred data has to be determined. The ERP-system is commonly not able to benefit from the high detail of data of the MES. A system needs to be chosen as the owner or primary care taker of the data for every master data. The owner system should usually be the one that works with the highest detail level.

4 CONCLUSION AND FURTHER WORK

As shown above ERP systems are not always up to the task to provide a sufficient IT support for shop floor processes. To overcome these limitations MES are used as a common solution. For manufacturing execution systems an integration framework has been established. Based on this an implementation approach has been introduced, which uses the total cost of ownership to select a scenario which fit the requirements best. With that a case dependent definition of MES and its integration with the ERP system is created. To ease the application of the implementation approach, the presented TCO model could be further detailed and the identification of the requirements could be supported by a typological analysis.

References

- Adam, D. 2001. *Produktionsmanagement*. Gabler, Wiesbaden.
- Bendeich, E. 2005. Produktionsplanung und Fertigungssteuerung mit erp und mes. *REFA-Nachrichten* 2, 4-10.
- Bendeich, E. and Strohmeier, S. 2004. Produktionsplanung und -steuerung aber wie? *Zeitschrift für wirtschaftlichen Fabrikbetrieb* 99, 538-543.
- Beuthner, A. and Fritsch, W. 2004. Lücken in der Feinplanung. *InformationWeek Special* 04,
- Gibbons Paul, L. 1997. What price ownership. *Datamation* 12, 88-94.
- Isa. 2000. *Ansi/isa-95.00.01-2000, enterprise control system integration, part 1: Models and terminology*.
- Isa. 2005a. *Ansi/isa-95.00.03-2005, enterprise control system integration, part 3: Activity models of manufacturing operations management*.
- Isa. 2005b. *Ansi/isa-95.00.04-2005, draft: Enterprise control system integration, part 4: Object models and attributes for manufacturing operations management*.
- Kletti, J. 2005. *MES - Manufacturing Execution System : Moderne Informationstechnologie zur Prozessfähigkeit der Wertschöpfung*. Springer, Berlin [u.a.].
- Loos, P. 1997. *Produktionslogistik in der chemischen industrie : Betriebstypologische Merkmale und Informationsstrukturen*. Gabler, Wiesbaden.
- Mertens, P. 2005. Gefahren für die Wirtschaftsinformatik - Risikoanalyse eines Faches. In *Wirtschaftsinformatik 2005* (Ferstl, O. K. and Sinz, E. J. and Eckert, S. and Isselhorst, T., Eds), pp 1733-1754, Heidelberg.
- Mesa. 2000. Controls definition & mes to controls data flow possibilities. *White Paper Number 3*, Manufacturing Enterprise Solutions Association, Pittsburgh.
- Reilly, K. 2005. Leitstände legen kräftig zu. *Computer Zeitung* 36, 14.
- Sap. 2004. Adaptive Manufacturing wird Realität. Walldorf.
- Scheer, A.-W. 1998. *Wirtschaftsinformatik: Referenzmodelle für industrielle Geschäftsprozesse*. Springer, Berlin [u.a.].
- Schmidt, L. 2004. Integration von mes- und erp-systemen. In *Integriertes Fertigungsmanagement mit manufacturing execution system (mes) : Mehrwert oder nur ein neues Schlagwort? ; Tagung Leonberg, 29. Und 30. Juni 2004* (Verein Deutscher Ingenieure. Kompetenzfeld Informationstechnik, Ed), p 168 S, VDI-Verl., Düsseldorf.
- Trebilcock, B. 2004. Is your erp supplier your mes supplier in the future? *Modern Material Handling*.
- Wannenwetsch, H. H. and Nicolai, S. 2004. *E-supply-chain-management : Grundlagen, Strategien, Praxisanwendungen*. Gabler, Wiesbaden.
- Wbf. 2005. Business to manufacturing markup language (b2mml).
- Wild, M. and Herges, S. 2000. Total cost of ownership (TCO) - ein Überblick. *Arbeitspapiere WI 1*,