Presented at the World Batch Forum North American Conference Atlantic City, New Jersey USA May 15 – 18, 2005



900 Fox Valley Drive, Ste. 204 Longwood, Florida 32779 USA 407-774-5764 FAX: 407-774-6751 E-mail: info@wbf.org www.wbf.org

How to use ISA95 part 3 for MES functional URS

Jean Vieille Consultant, Psynapses Network 10, rue du Stade Dompierre-les-Tilleuls, 25560 France +33 6 11 62 52 61 jean.vieille@isa-france.org

KEY WORDS

ISA95 part3 – MES – URS – System adequacy and selection

ABSTRACT

Shop floor activity execution control encompasses many management functionalities, addressing several types of manufacturing operations from the receipt of raw material to the shipping of finished goods, from production itself to equipment maintenance through inventory movements and material quality tests. These activities fall under different responsibilities, though they must operate collaboratively under the business management directions. When it comes to specifying such a global control/MES system, the user requirement specification (URS) may appear a difficult task, leading to confusing and complex documentation, and eventually project failure.

Since the ISA95 standard initial scope just covered B2M information flows between Production execution and Business systems, the SP95 committee had to pay attention to the execution layer functional duties. The first valuable contribution to sort the chaos within this large scope of functional requirements was the "11 functions" MESA model. Part 3 of the ISA95 standard defines a more elaborated functional model particularly suitable for URS development. This presentation describes a formal method for defining MES URS and system adequacy analysis based on ISA95. This method was used for very different projects such as: defining a global MES template for a major tire manufacturer; implementing a fully integrated Control/MES for an animal food company; specifying MES URS for a paper facility.

PAPER

1. Introduction

1.1. What's wrong with MES?

When it comes to deal with MES, no one has the same feeling about it. From the manufacturing viewpoint, considering SOP enforcement and data collection as the core MES duty, to the business viewpoint of managing inventory and fulfillment from the shop floor. It is this kind of perception divergence that needs to be clarified. The differences come from two facts:

- Manufacturing involves both resources/product handling/processing and business/execution management, which are very different perspectives of the same subject
- Two communities are involved in addressing it: IT/Business "managing" community which hardly understands actual manufacturing control constraints and needs and the operational "executing" community which doesn't comprehend business mind and vision

From that situation, functional responsibility questions and integration concerns appear among both the systems and people involved in manufacturing business and execution management. This makes difficult to structuring requirements, resulting in high challenging assessment process that takes a unique mixture of skill sets and years of study and practice.

ISA95 and MESA exist precisely to help the emergence of common understanding and best practices sharing.

1.2. What's wrong with URS?

The first step to build a system is to define the requirements it is intended to address. While they are sometime expressed in very broad and blurry way, a precise definition of the actual needs is a key factor for succeeding in implementing a somewhat useful and productive system.



As an example, the pharmaceutical industry naturally places URS at the very top of the GAMP cycle. However, this diagram shows an odd multi-link between both URS (under customer responsibility) and FS (Functional specification, under vendor/integrator responsibility) to System Acceptance testing. Project quality management has to guarantee URS coverage through a URS/FS coverage matrix mapping URS topics to FS modules.

URS are not qualified by themselves, but indirectly by agreeing on the FS solution addressing each individual requirement.

Other project management approaches have similar concern in addressing consistently URS.

Advanced process control URS methodologies can make extensive use of ISA88 to ensure the consistency between URS and FS relying on a common robust functional framework. FS becomes a more elaborated URS instead of being a brand new solution-oriented thinking extrapolated from a hardly maintained one-shot verbose literature. ISA95 part 3 (current draft 21) should play the same role for MES, allowing a more straightforward mapping between URS and FS.

1.3. ISA95 part 3 functional framework

ISA95 provides extensive common sense models, explanation and terminology to discuss and address manufacturing execution management in a more consistent and consensual way than ever. While there are still discrepancies within the standard, a consistent informational and functional framework takes shape.



The part 3 activity management model can be applied to different Manufacturing Operation Categories (MOC) such as those appearing in the PRM functional model. The standard suggests Production,



Quality, Maintenance and Inventory as commonly defined categories.

Also, the standard defines supporting functions, which specify common infrastructure requirements.

The result is a tri-dimensional functional framework (Manufacturing Operation Categories, Core Activities and Supporting Activities) that can be used for URS as well as for solution mapping.

1.4. Functional hierarchy



The proposed method for URS development is *process based* as shown on the figure bellow

Between business planning and work execution processes and management responsibilities, MES concepts are about inserting an execution management layer between business oriented enterprise planning and execution oriented enterprise manufacturing. Thus, business processes are split or shared between business and execution responsibility domains.

2. Methodology overview



The methodology starts from defining actual Manufacturing operation categories as execution Responsibility sub-domains to drive requirement data collection. Beside this decisive start of the organizational and process based functional architecture design, resource modeling gives substance to the project and support for specific requirements by setting up the current facility capabilities.

Subsequently it defines Business Processes and identifies the tasks they drive. Business processes

define the situations and scenarios that are involved for having a system achieving its' objectives.

Collaboration between MOCs within execution responsibility domain and with Business responsibility



sponsibility domain and with Business responsibility domains is highlighted, providing a comprehensive input for likely integration requirement.

Tasks are characterized, classified and consolidated to build up or to comply with an Enterprise MES functional core system.

Finally, the actual solution capabilities are compared to the functional framework to conduct adequacy assessment and URS/FS mapping.

The facing figure shows the URS development sequence.

3. Methodology hints

3.1. Phase 1: Technical modeling



Technical modeling describes the target facility. It includes 5 steps to define MOCs and resources (both MOC specific and shared).

Resources can be defined independently, while working segments (a generalization of "Process segments) are built from these resources.

This initial step is mainly about the system configuration that can dynamically change over the time than the system design itself.

3.1.1. Step 1.1 Manufacturing Operation Categories

ISA95 Part 3 defines the following Manufacturing Operation Categories (MOCs): Production, Quality tests, Maintenance, Inventory control. Other or different MOCs can be defined. Example: Inbound, Outbound logistics, Internal transfers, Tooling, Cleaning... Sometime several MOCs can be defined for a given activity domain. For example, bulk production and packaging can be defined as specific MOCs instead of being part of the same "Production" MOC.

The key words behind MOCs are Execution Responsibility and Typology.

3.1.2. Step 1.2. Material Resources

At the URS level, only Material classes are identified. Material classification allows specifying segment input/output, filter information and associate specific functionalities or processes, etc... Actual materials are initialized at the system configuration and will be dynamically managed by the user. The average number of material definition can be stated here. Material Properties can be defined at a further development stage.

Classification is multidimensional and shall be consistent with business modeling options

3.1.3. Step 1.3. Equipment Resources

Equipment can be defined according the ISA95 physical model (though this is not mandatory if the Enterprise already has one). All main (schedulable) pieces of equipment are identified, classified and positioned within the physical hierarchy. Equipment Properties can be defined at a further development stage.

Equipment identification and classification is the main input for work segmentation and system user access definition.

3.1.4. Step 1.4. Personnel Resources

At the URS level, only Personnel classes are identified. Personnel classification allows specifying segment required qualification, system accesses, etc... Actual persons are initialized at the system configuration and will be dynamically managed by the user. The total average number of individuals can be stated here. Personnel Properties can be defined at a further development stage.

Classification is multidimensional and shall be consistent with business modeling options

3.1.5. Step 1.5. Working segment

Working segments define the actual work capabilities of the facility and associate the corresponding resources. *Note: The standard currently only defines Process segment, but the powerful Product/Process segment concept will certainly be soon generalized to other manufacturing operation categories.*

Segments are hierarchically defined from a very broad perspective (the entire factory making a particular brand of products) to the smallest processing capability (a single ISA88 phase), though the most useful granularity may correspond to ISA88 unit procedures. They are part of the general control and automation architecture.

3.2. Phase 2: Business processes

Business processes define the highest functional requirement level. They illustrate situations and tasks activation scenarios, they could be

- Manual, semi or fully automated,
- Hierarchic: High-level processes activate lower level processes; elementary processes are tasks

Example of BP can be: Bulk material receipt by truck, by train, Palletized material receipt, On-receipt, on-line, on-shipment analysis, Bulk Product shipping, Internal bulk transfer, New orders processing,



Order cancellation from execution, from business, Unsolicited order reporting, Order completion reporting, Process improvement suggestion processing, Material lost reporting...

Business processes can be compared to manufacturing processes: ISA88 Recipes (BP) activating EPEs (tasks). An example is shown on the facing figure.

This example makes use of BPMN (Business Process Modeling Notation), a high level, simple to learn and use workflow description language that can be mapped on design oriented languages (in case of computer controlled processes)

such as BPEL (Business Process Execution Language), BPEL4WS (BPEL for Web Services), BPML (Business Process Modeling Language).

Business processes can be categorized at will to help extensive and structured requirement gathering.

- Execution management processes deal with work organization and execution.
- Resources management processes deal with resources rather than work execution
- Operation management processes deal with operation performance, including dashboards, performance indicators, general activity reports. They are not linked to work orders or resource control
- Repository synchronization processes aim to maintaining consistency between configurations databases in all systems involved in previous processes. The need for such processes largely depends on business management policies and supply chain dynamics.

Business processes can be partially or fully automated, they can be manually controlled as well.

Business processes can be MOC specific, generic for several MOCs or shared by several MOCS. For example, on-line QC tests process can involve tasks within Production and Quality test MOCs.

Business processes can be confined within Business responsibility / system, within Execution responsibility / systems, or they can cross the Business/execution responsibility / system boundaries.

Thus business processes are the primary input for identifying interfaces and corresponding messages and transactions.

3.3. Phase 3: Tasks



3.2.1. Task definition overview

The business process definition phase highlights the tasks involved in their execution. A list of tasks can be established as the basis for functional definition. These tasks are organized following the ISA95 part 3 activity model under each MOC.

The next steps will elaborate, refine, consolidate and classify this task list. Consolidation attempts to reconcile similar tasks into generic, less numerous task classes. From the optimized task list, business

processes are amended to match this more global functional modular definition.

Task definition and business process alignment are executed simultaneously and iteratively until satisfaction. Doing this exercise the first time will end up with a consistent set of hopefully reusable functional components. Next projects will make use of these components as much as possible. They will improve and complete them to form the Enterprise MES components core system.

3.2.2. Step 3.2 Description

Task description includes two dimensions: Characterization that defines usage attributes and Justification and Functional description that defines the services to be provided.

Typical characterization attributes can be:

- Physical level defines the position of the task in the physical/decisional hierarchy
- Manufacturing Operation Categories restrict the task applicability to particular manufacturing operation categories
- Segments restrict the task applicability to particular work/working segments
- Technical constraints define the task environment and condition of usage
- Dependences indicate that a task execution depends on the existence of other tasks. This is used to justify apparently non value adding tasks
- Task style defines the type of functionality the task achieves (real time or transactional control, data storage, knowledge management, data analysis, simulation and modeling, ...)
- Responsibility specifies if the task is under the responsibility of business or execution
- Users specifies the personnel classes who have access to the task
- Justification gives technical and financial reasons for implementing the task and associated implementation priority
- Information defines the consumed / produced / manipulated / presented information based on ISA95 part 1/2

Functional description will actually explain the expected task behavior in normal and exceptional operation conditions





The MOC and process oriented specification produce a basically inconsistent list of tasks. The last step is to sort them out in order to have a reduced, consistent, reusable list of tasks hopefully mapping the growing up Enterprise MES component core-system.

This object-based classification will greatly simplify and harmonize execution processes by propagating best practices and questioning unjustified specifics.

For example, multi-MOC similar tasks can be

 harmonized between MOCs (identical, but separate); example: dispatching work instructions / Microsoft Word

- kept specific by MOC (they are different); example: work definition / AutoCAD
- or shared between several MOCs (the same task instance address the functional needs for several MOCs); example: managing personnel information / HR management system

3.4. Phase 4: Solution selection

Selecting the solution may follow the following approach.



3.4.1.Step 4.1 Pre-qualification

The purpose of this step is to pre-select a limited list of the best possible solutions based on non-functional criteria (company's and product records, local support, integration capabilities, technology...)

3.4.2.Step 4.2 Functional adequacy

The purpose is to match the functional requirements (consolidated tasks) with the solutions capabilities.

When considering actual solutions against functional requirements, one has to consider the different aspects of the solution



- Inherent capabilities providing embedded solution functionalities that can map onto one or several tasks
- 2. Integration capabilities that can leverage inherent capabilities to fulfill the requirements. Integration is what can be done within the system to develop interaction between standard modules without impacting the solution integrity

3. Specific development that are needed to complement the solution inherent capabilities (1) and integration artifacts (2).

- 4. Existing applications that can still operate in the target solutions
- 5. Manual procedures that can be applied instead of a software implementation

The possibility for a particular element of the solution – an application component - to address more than one task requirement is assessed against technical constraints, information consistency, user interaction, and applicable Manufacturing Operation Categories.

This is an objective process that can help to reduce the number of possible solutions by eliminating those that inherently don't cover enough requirements using methods (1) and (2)

3.4.3.Step 4.3 Mock up

This step will implement different scenarios involving a set of tasks and system integration features to test the possible remaining solutions.

The result is a shorter list keeping the only suitable solutions in term of integration and interfacing capabilities and actual feasibility.

This will improve the subjective perception of the solution, but it won't assess its behavior in normal operating conditions and scalability.

3.4.4.Step 4.4 Select and order

This is the final step where technical considerations quit for financial negotiation

3.4. Phase 5: Implementation synchronization

The URS stage is not really worth carrying out when:

- Only initial system implementation is considered
- URS is forgotten as soon as the first functional specifications are released.

However if:

- The project outcome needs to be controlled,
- Enterprise best practices and know how are considered as a valuable asset,
- System lifecycle has to survive commissioning,

Then system independent URS are unavoidable, and their maintenance is a critical issue.

- URS shall be in sync with actual implementation: unspoken requirements showing up during the system implementation and commissioning have to be included in the URS.
- Each new or modified requirement shall be engineered at the URS level to maintain the consistency with the core-system, to address issues from a broader perspective, and to make available at the enterprise level the corresponding improvements
- User and solution integrator must agree on the documentation form and be committed to use it as the highest level contractual interface between all project actors

4. Conclusion

MES implies a clear perception of manufacturing control and planning.

Conducting successful MES projects and building flexible, evolving solutions is highly challenging. It can only be achieved thanks to a high enterprise maturity level and a full commitment to ongoing improvement.

The ISA95 standard brings a common vision between business and execution. The whole standard is an excellent tutorial for understanding MES functions and involved information from a universally shared perspective. Part 3 is particularly useful to structure URS. The process and MOC based requirement analysis improve the URS completeness and transparency.

The proposed methodology follows the spirit of the standard by:

- Encouraging responsible user involvement in the system definition,
- Helping mutual understanding between project actors,
- Being independent of the technical solution, allowing full deployment flexibility and system evolution