Dynamic Tableau-de-Bord: concept, tools and applications

*J. M. Neta, P. A. Silva, B. Ribeiro, J. M. Mendonça**

INESC - Instituto de Engenharia de Sistemas e Computadores

Rua José Falcão, 110 - 4000 PORTO - Portugal

T: +351 2 2094314/7 F: +351 2 2008487

e-mail: jneta@inescn.pt

Abstract

In order to more accurately evaluate the performance of a manufacturing enterprise, new concepts are needed for measurement and evaluation. The evolution of concepts in this area clearly indicates that non-financial indicators are the relevant information at all stages in manufacturing.

These indicators must be always available, up-to-date, relevant, easy to understand, simple to calculate and anyone executing productive tasks should be committed to monitor and react to them.

However, measuring all the needed features is a very complex and time-consuming task. Inevitably, it must be supported by new software tools.

Since Manufacturing Execution Systems are becoming a common solution in the majority of Portuguese companies, an adequate architectural system approach must be found in order to sustain new developments as these trends become more and more clear.

Within the scenario above, work is being undertaken in Real-I-CIM, an ESPRIT project which aims to develop a tool-set for building customer-driven manufacturing management solutions. Within the scope of the project, a novel concept has been introduced - the Process Conditioning Monitoring, emphasizing that continuous process monitoring is essential to build a constantly improving performance measurement system.

The Dynamic Tableau-de-Bord application, a key element in this context, will implement a software platform for continuous and reactive measurement of performance related parameters.

* Faculdade de Engenharia da Universidade do Porto - Departamento de Eng^a Electrotécnica e de Computadores - Rua dos Bragas - 4050 PORTO - Portugal

T: +351 2 6103359/60 - Fax: +351 2 6103361 - e-mail: jmm@adi.pt

1. INTRODUCTION

1.1 Competitive production and performance measurement

The competitive enterprise of the future has to strive for continuously ensuring the improvement, in terms of value, of their products and services. This process of continuous improvement, frequently linked with novel organizational paradigms and manufacturing strategies (cells, cooperative work, Kaizen, TQM, etc.) can and should be quantitatively assessed through adequate performance measurement.

Performance measurement is now recognized of paramount importance regarding both more traditional aspects, such as cost, quality and time, and crucial emerging factors, such as precision of delivery, reactivity, flexibility and time-to-market. The acknowledgement of these emerging factors brought in new requirements, that cannot be met by present systems and solutions for manufacturing management, and that played a significant role in the motivation for the Real-I-CIM* project.

1.2 Key Performance Indicators

1.2.1 Measurement of Quality Performance

There are several definitions of quality. The most popular are [2][4]:

- “conformance to requirements”,
meaning, conformance to engineering specifications;
- “fitness to use”,
meaning, conformance to user requirements;
- “customer satisfaction”,
giving a final emphasis in customer demands fulfillment.

Quality assurance means continuous improvement aiming at the systematic elimination of rejection causes. To measure quality, several indicators related with different areas of the company are used. The most widely used are:

- Vendor quality;
establishing special relationships with the suppliers in order to increase and control the quality of incoming raw materials is one of the major guidelines; incoming quality calculations can be performed by counting the number of rejections (in lots or units) for each supplier and product and the percentage over the total quantity;
- production quality;
it can be assessed through rejection rates analysis and non-conformities analysis; rejections mirror the percentage of the production classified as defective; non-conformities are the products quality characteristics that contributed to the product rejection;
- data accuracy;

* Real-I-CIM is the ESPRIT 8865 project undertaken by INESC (Porto-Portugal), TECNOTRON (Lisboa-Portugal), Fhg-IPA (Stuttgart-Germany), AESOP (Stuttgart-Germany) and DICONSLT (Munchen-Germany). The consortium has two pilot-sites: GROWELA (Maia-Portugal) and BMW (Lohhoff-Germany).

data accuracy is ensured through inventory accuracy checks, BOM and routing checks and forecast accuracy checks;

- effectiveness of preventive maintenance;

applying a preventive maintenance program can improve quality assurance by increasing machine reliability; the most common indicators are the down-time over a time range, the number of machine unscheduled interventions and the average time between breakdowns;

- quality costs;

quality costs are an essential measure of the efficiency of the company Quality System; usually, they are divided in: non-quality costs which are split up into internal costs (scrap, rework,...) and external costs (warranty charges, client devolution, etc....); appraisal costs, incurred to discover the conditions of the product (inspection costs, evaluation of stocks, ...); and prevention costs incurred to prevent non conformities, i.e., minimizing the two previous categories (training, auditing, etc.).

1.2.2 Measurement of Maintenance Performance

Maintenance can be defined as “a set of actions or operators that maintain or reestablish a system state or ensure a given task” [1].

To ensure such functions in a maintenance management system, a number of performance indicators is needed:

- Time analysis.

Equipment availability and productivity are of prime importance in manufacturing companies, and simple and powerful steady-state equipment indicators can be obtained through MTBF (Mean time between failures) and MTTR (Mean time to repair). From these, it is usual to derive the reliability rate (λ), the repair rate (μ) and the equipment availability (also called the up-time ratio).

The equipment availability is obtained through a simple calculation, in percentage format:

$$\lambda = \frac{1}{\text{MTBF}}, \mu = \frac{1}{\text{MTTR}}, \text{Availability} = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}}$$

Maintenance interventions and manpower resources can be analyzed with specific time indicators. These enable splitting maintenance time into operators time, equipment time, testing time, corrective maintenance and/or preventive maintenance times. These values are necessary to plan periodic preventive maintenance actions, to schedule operators allocation according to their availability and performance, to continuously improve maintenance procedures and to calculate maintenance costs.

All indicators referred to above are analyzed on a equipment-to-equipment basis, and can be consolidated to equipment families, work centers or the entire plant.

- Maintenance costs.

The maintenance policy is usually closely linked to cost analysis. The maintenance policy should always evaluate the current costs of maintaining the equipment in a preventive maintenance program, repairing, purchasing surrogate equipment, etc.. One good reason to continue using some financial related costing is that the maintenance department often strongly impacts on the company's global costs. Precise budget and deviations analysis are thus very important.

These costs indicators can be used for budget forecasting, budget execution, to set the preventive maintenance level, to decide on subcontracts and acquisition of external manpower, to decide whether to repair or to replace equipment, etc.

- Global performance indicators.

Some special ratios are important for evaluating the maintenance group performance.

The number of reported breakdowns is important to measure the disruption frequency versus production. These numbers can be extended to determine the weight of corrective maintenance actions over preventive ones.

The costs ratios between intervention costs and total breakdown costs are also important and the occupation time of the maintenance team over their capacity is frequently used.

All such ratios are analyzed over pre-defined time intervals.

1.2.3 Other Important Indicators

Other measures may complement those presented above, covering for other aspects in manufacturing. A selection of those found more important is given below.

- Delivery Performance and Customer Service.

The key areas of interest are vendor delivery performance, production schedule adherence, order and schedule changes, customer service level and lost sales analysis.

- Measurement of Process Time.

It includes the break-down of production times into the various components of the cycle time. Cycle time is the time required for manufacturing a product. It includes the time spent in all the required equipment, queuing, transportation, inspection or other activities.

$Ct = pt + mt + wt + it + st$, where

Ct - cycle time (total shop-floor manufacturing time)

pt - processing time (time required to actually perform the operations)

mt - move time (time spent in movements)

wt - wait time (time spent in queues or buffers)

it - inspection time (time required for quality inspection)

st - setup time (time required for preparation of production equipment)

The goal in measuring cycle time is to increase the importance of the value-added activities versus the non value-added ones.

Other commonly used indicators are the D:P ratios (D stands for the lead time offered to customers and P stands for the production lead time, which is the total time required to do all the required sub-tasks including purchasing raw-materials), the material availability, the distance covered by material movements during production and the customer service time.

One should therefore, add to the Ct the time spent in purchasing raw-materials, or in ensuring in any way the material availability and the customer delivery time.

- Measurement of Production Flexibility.

It aims to measure the degree of adjustment to new customer order specifications. It can only be measured controlling indirect indicators such as the number of different parts in the BOM, the percentage of standard, common and unique parts within the BOM, the number of different

production processes, the position within the production processes where the products are differentiated, the number of levels in the manufacturing BOM, the introduction of new products, the cross-training of production personnel and the comparison of production output and production capacity.

- **Measuring Social Issues.**

Here one tries to control some of the most subjective and hard to measure indicators. These can be found through the analysis of the morale of the work force and teamwork, the involvement of employees, leadership ability of managers and the company's commitment to safety and environmental procedures.

1.3 The Real-I-CIM project

The novel concepts behind REAL-I-CIM build on the innovative integration of a few strong, i.e. proven, widely accepted and simple, ideas which are laid out below.

- it aims at providing decision-making for factory management at different levels supporting:
 - improved productivity (costs, time, etc.),
 - improved quality,
 - high flexibility and reactivity to internal and external perturbations;
- it allows user-driven customized solutions to be built using:
 - an integrated tool set based on market standards, both business and IT standards,
 - problem-oriented analysis and implementation methodologies enabling organizational changes and stressing the management of change;
- integration with the existing environment is achieved through an open architecture and infrastructure and a core information model which allow an high degree of interoperability, and therefore adaptable and extendible solutions can be built and interfaced.

The results of REAL-I-CIM can be looked at as "a tool-set and an implementation methodology for building advanced shop-floor management systems".

All the applications developed within the ESPRIT 8865 (RIC) project will be bundled in PROFIT* (Production Optimization through Flexible Integrated Tools) as full commercial products.

1.4 Process Condition Monitoring

1.4.1 The PCM Roots

The RIC Process Condition Monitoring (PCM) concept builds on more classical concepts, specifically those of Machine Conditioning Monitoring, Process Monitoring through SCADA (Supervisory Control and Data Acquisition) Systems and Continuous Improvement (Kaizen engineering) [1].

* PROFIT is a consortium among R&D partners and software-houses. These are INESC (Porto) and Fhg-IPA (Stuttgart), for the R&D solution providers, TECNOTRON (Lisboa) and AESOP(Stuttgart) as the marketing partners.

1.4.2 The PCM Control Loops

The RIC PCM concept integrates real-time monitoring with advanced control at a lower (on-line/shop floor) level, and integrated performance monitoring and evaluation with intelligent data analysis and knowledge extraction at an higher (off-line/engineering) level.

For monitoring and controlling purposes the PCM tool set includes modules performing shop-floor data collection, data logging and simple data analysis and interpretation, within the scope of the classical quality, maintenance and WIP/order tracking functionalities. These modules collect and process relevant information and offer specific decision support and controlling functions within the scope of the specific users environment.

These facilities will be available in a readily configurable manner as to build PCM loops with integrated functionality. These local real-time information and decision control loops will allow higher level integrated PCM client applications to source information from the database server providing aggregate information and decision support.

The modules/functionalities provided are now presented in more detail, grouped in three main streams: quality, maintenance and dynamic tableau-de-bord.

1.4.3 Quality Tool-set

One of the goals of the RIC Quality Tool-set is to help manufacturing companies in the implementation and maintenance of their Quality Systems. The RIC Quality Tool-set specification was based in the ISO 9000 and FORD Q101 standards (see **Table 1-1**), the Quality standards adopted in most industrial sectors.

Another goal of this tool-set is to provide an adequate infrastructure embedded in all relevant areas of the quality function, i.e. allowing quality information gathering and availability at the different areas of the company.

Finally, this tool-set should also be easily customized to the organization needs since the procedures of a Quality System are highly dependent on the specific organization.

These goals also comply with the encompassing PCM concept, as the RIC Quality Tool-set provides a set of highly flexible and easily customized Quality tools.

Table 1-1: ISO9000 requirements versus RIC requirements

ISO 9000 Requirements	RIC
Management responsibility	
Quality system principles	
Internal quality audits	
Quality Costs	✓
Contract review	
Design control	
Document Control	✓
Purchasing	✓
Purchaser supplied product	✓
Product Identification and Traceability	✓
Process Control	✓

Inspection and testing	✓
Inspection, measuring and test equipment	✓
Inspection and Test Status	✓
Control of non-conforming product	✓
Corrective Action	✓
Handling, storage, packing and delivery of materials	
Quality Records	✓
Training	✓
Servicing	
Statistical Techniques	✓

1.4.4 Maintenance Tool-set

The Maintenance application includes flexible and customizable maintenance management functions capable of full integration in a real-time interactive production management system. The following features are provided (see **Table 1-2**):

- easy integration with company legacy information systems;
- grouping, coding and description of the company manufacturing equipment;
- data collection and shop-floor interaction using appropriate terminals;
- fast and simple answers to the Maintenance department requests and, as a consequence, to the Maintenance operators;
- continuous information updates for historical purposes and better management of Maintenance interventions;
- maintenance actions planning in agreement with the Production Master schedule;
- improved planning through the optimization of both Maintenance and Production times;
- inter-department time cross-analysis to improve the performance of their actions and to support improvement procedures;
- maintenance costs analysis and evaluation of the Maintenance policy (adequate balance between Preventive and Corrective Maintenance is searched for);
- study of the functional behavior of the equipment in their different life periods;
- extraction of indicators for the performance evaluation of industrial processes and management procedures;
- exception handling at the shop-floor level.

Table 1-2: Available versus specified functions in RIC maintenance application

Module Name	RIC
Corrective Maintenance	✓
Preventive Maintenance	✓
Predictive Maintenance	
Time Analysis	✓
Maintenance Costs	✓
Maintenance Statistical Analysis	✓

Document Control	✓
Training	✓

1.4.5 Dynamic Tableau-de-Bord

The dynamic Tableau-de-Bord is an application for handling performance indicators and for defining a control board in a management context, using a very graphical user interface for data exploring. It can thus be used as a decision support tool.

Indicators are grouped in decision boards, which are called *Tableau-de-Bord*, in fact, sets of indicators in a management context which enable a synthetic view over a system. The indicators in a Tableau-de-Bord can be correlated, consolidating information and deriving more elaborate indicators [5]. They must be animated using powerful graphical facilities giving them an enormous expressive power.

The big difference to conventional reporting tools is the dynamic presentation facility. In this sense, indicators are dynamically updated as the system information is.

1.4.6 Other Modules

There is also an extra PCM level [1], that includes knowledge-based (KB) tools. Advantage will be taken of the historical recordings with information from lower levels, to extract, organize and improve the knowledge on the process, allowing reference values, corrective actions and controlling strategies to be derived and deployed.

One of the basic functions in RIC, yet one of the most important ones, is the Alarms system. This application enables the registration and acknowledge of exceptions during the operation with RIC applications.

1.5 Manufacturing Execution Systems

1.5.1 Information and Control Model

All information handling problems within a company are structured in accordance to a hierarchy. In order to support the new manufacturing techniques or novel forms of organization, there must be a very well integrated solution addressing all levels of information needs.

Roughly, problems can be addressed in the following categories [7][11]:

- Planning level, which includes functions like forecasting, budgeting, logistics, order management and MRP (materials requirements planning); more advanced systems already include MRP II facilities (manufacturing requirements planning).
- Control level, that includes functions such as process, machine and equipment control. It encompasses equipment interfacing, communications and data collection/acquisition systems.

For these two levels, the market presents a wide variety of solutions. Some are general purpose, some very process optimized. However, the gap between those information levels is immense. Production dispatch is only accomplished through manual records or theoretical forecasting. Actual plant information has to follow tricky paths to go from shop-floor to the planning department. There is actually no real-time reactive ability in the system, since information from the production line does not reach in time the planning area.

Manufacturing Execution Systems (MES) appear to fill in the space between planning/business systems and their related process control systems [9].

1.5.2 The MES Layer

A MES is the solution for building a consistent real-time information infrastructure. A good system in this layer enables the company to implement continuous improvement, JIT, customer oriented production strategies and other modern production concepts. The core functionalities provided by RIC are at the MES level.

The major guidelines for the positioning of a MES action are [8]:

- plant management, including functions to manage resources, capacity scheduling, maintenance and product distribution;
- plant engineering, enabling company-wide document management and process optimization; it manages routing plans, schedules and implementations, keeping written records at all stages;
- quality management, implementing the inspection and analysis procedures for Total Quality Management; it includes Statistical Process Control (product or process quality reporting, supply inspection and rating, certification reporting, failure analysis, identification and traceability, Pareto analysis and SPC charts), quality costing and inspection, measurement and test equipment management;
- supervisory control and monitoring, including process management and shop-floor data collection.

1.5.3 MES Performance Measurement Functions

Specific high-level indicator functionalities are needed within a MES environment. Indicators are useful at any level but specially at this one, for three different reasons:

- all real-time control operations are held at this level;
- this is the closest point to both planning information and shop-floor data collection;
- indicators should be available to any operator.

One of the most important RIC goals is the construction of an application which embodies these features. That application is called Dynamic Tableau-de-Bord (DTdB) and will be now presented.

2. THE DYNAMIC TABLEAU-DE-BORD APPLICATION

2.1 System Integration

In order to implement the DTdB application, indicators must have some physical meaning within the RIC system. Indicators are delivered by system entities called *data servers* (see **Figure 2-1**). These entities provide high level *services* that implement the indicators, among other features [2].

The main goals of data servers are to provide a fully integrated and unified information model, suitable to every application in the system, and to provide integration with other systems (the so called legacy systems, such as MRP, LIM and managerial type applications).

Therefore, a data server is defined as a source of structured information concerning a certain domain (Quality Control, Maintenance Management, Reactive Scheduling, Simulation or Shop-Floor Monitoring).

2.1.1 Data Servers Conceptual Architecture

RIC applications have a typical Client/Server architecture which means that each one of them has a data server and, at least, one client. Typically, the data server component will be responsible for all major data processing features and the client side will deal with the user interaction.

The following requirements concerning data access were identified for RIC applications:

- data should be encapsulated in data servers, according to its information domain;
- data access has different levels of abstraction, from plain *database access* through to the complex *business rules*;
- moreover, these business rules usually remain unchanged, even when the user interface and/or the data source structure or format are changed;
- the format of the data source must be transparent to the applications;
- the data servers should rely on a common distributed infrastructure.

Figure 2-1 - Functional levels of a data server.

A data server has thus two levels of functional abstraction: business rules, which implement core services related with its information domain, and basic data access functions (see Figure 2-1). It supplies a set of services encapsulating the information model to the applications. Since this model usually changes in each system installation, the proposed architecture will promote reuse.

A data server should provide data access to their clients, services access over data by encapsulating procedures in one common structure, and a mechanism to trigger shared events (a signal sent to client applications upon occurrence of certain conditions) notifying client applications.

2.1.2 RIC Architectural View

The concept of Client/Server computing has been evolving and different implementation models have been proposed.

In RIC, information was split into several domains which exist in a decoupled way. Therefore, it is easier to develop and maintain applications which are built in different platforms, by different partners and for different end users.

For these reasons, the integration problem has been divided into the following categories:

- the information domain problem, which is dealt with by defining as many data servers (logical servers, not necessarily physical) as different domains in the specification for a specific company;

- the client/server architecture, which was developed as the natural upgrade from two-tiered systems into three-tiered ones [10].

According to this description, RIC applications may have the following type of connections [3] (see Figure 2-2):

- SQL connections to their data source, since SQL has been the well accepted standard for database interfacing and relational databases are, most of the time, the widely used data store;
- connections to the RIC system servers, which are message oriented applications and, for that reason, they rely on a common message transmission system.

Besides database servers, RIC applications may have the need to retrieve data from foreign systems (MRP's, Data Collection Systems, etc.). In each case, there is a distinct interface to those systems.

Figure 2-2 - Logical and physical view of RIC data servers.

The system thus evolved according to the following needs of encapsulation:

- data access, so that different kinds of data sources and structures could be hidden from the users and applications point of view;
- business rules, so that the procedural description of value added services could be hidden from the application itself.

A major goal was to provide service orientation instead of data orientation. Since not all RIC applications could easily cope with this goal, the Dynamic Tableau-de-Bord application was used as the test bench for these novel concepts.

Furthermore, the RIC architecture was designed according to an object-oriented paradigm. Each object comprises the following components [2][3]:

- the client;
- the ORB - the object request broker;
- the object implementation.

The ORB layer is responsible for all the mechanisms required to find an object's implementation for the request, to prepare the object implementation to receive the request, and to communicate the data accomplishing it.

Clients have no knowledge of the implementation of the object, its location, or the exact ORB that manages that object.

An object implementation provides the semantics of the object, usually by defining data for the object instance and code for object's methods. Often, the implementation will use other objects or additional software to implement the behavior of the object. In some cases, the primary function of the object is to have side effects on other components that are not objects.

A variety of objects implementations can be supported, including separate servers, libraries, a program per method, an encapsulated application, an object oriented database, etc.

The many tiered architectures have advantages over classical two-tiered systems:

- scalability, that is, the size of the problem can always be overcome with physical distribution of processing resources, maintaining all the major interface features;
- performance, since it has the system load more balanced and distributed;
- reusability, since objects encapsulate details that only concern to a certain context in the hierarchical structure; in this way features can be more easily added or changed without major changes in the whole system;
- specialization, that is, tasks are better divided between client and server; clients only deal with user interface problems.

Classical architectures are almost always dependent on SQL engines, and all the processing is bundled within them. The database server process is a bottleneck, although some engines already have distributed features. Nevertheless, this implies that large investments must be made in hardware and software.

When an application wants to access a service provided by a server that is not a SQL engine, an interface must be provided at the client level. With many tiered architectures, this problem is solved at a lower level.

Some disadvantages may arise. When an application needs to manage very large amounts of data which do not require really heavy processing, with low complexity, many-tiered systems may be redundant, and overheads may affect the overall performance. This happened with all major functions for Quality and Maintenance tool-sets, since they were very data oriented. For this reason, those two applications were kept on a so-called classical architecture.

2.1.3 The DTdB Implementation Process

The prime goal of the implementation process was to test all these innovative concepts. So, the new directions were:

- to test and validate the advantages of three-tier architectures over two-tiered ones;
- to present concrete solutions to the architecture transition problem;
- to study new development tools which are more suited for deployment according to the new concepts;
- to study new solutions for data storage and comparing its performance and ease of use with present technologies.

Therefore, the DTdB application used:

- object-oriented methodologies for design and implementation;
- dedicated data servers which encapsulate application specific indicators;
- conventional databases (although one of the main features of this approach is the possibility of connecting with different types of data storage, such as state-of-the-art object-oriented databases);

- an object request broker for object distribution;
- a very graphical user interface tool over a windows-based system (X-Windows or MS Windows).

The entire DTdB application already has a prototype available. Its deployment was devised in three stages:

- Non-configurable programming interface.

The programming interface is static and only available for the application programmer. The effort is put in complying with the user interface and system requirements. Therefore, data servers provide performance indicator services, the DTdB is able to connect to them and the results are displayed according to the user interface requirements.

- Configurable programming interface.

The user can redefine a Tableau-de-Bord in a WYSIWYG (what you see is what you get) manner. There are no further major improvements, at this stage.

- Deriving new indicators.

A programming rule-based language is provided in order to put some programmable processing facilities at the client side. New indicators can thus be derived without re-programming the system data servers.

2.2 User interface

As it was already stated, a very graphical user interface is a must. The indicators must be presented to the user in the following manners:

- table or grid formats;
- graphical charts.

Charts have several types, according to the final goal:

- line or bar charts, that are more suited to see trends, indicator evolution across time or any other category, etc.
- pie or scattered bar charts, which are more suited for comparisons among several categories;
- three-dimensional charts, when analyzing indicators with three or more categories.

For specific reasons, combinations of these techniques can be used to produce special type of graphs such as ABC or Pareto-type graphs and Gantt charts.

The system must be capable of presenting results:

- On the screen.

The dynamic requirement imposes some animation which will only be relevant on screen presentations.

- On paper, through a printer.

That is to say, it must have conventional reporting facilities.

The definition of a Tableau-de-Bord must be simple and easy to understand. The indicators set structure will evolve with time, so a new Tableau-de-Bord will have to replace older versions.

The connection to the system infrastructure should be intuitive, since indicators are set upon system services. The programming skills of the system users should not be a relevant pre-requisite to define new Tableau-de-Bord.

3. CASE STUDY: THE GROWELA PILOT SITE

Growela has been developing, along with an industrial consulting company (Roland Berger), a set of indicators that should help them to monitor several internal processes performance, to define intervention areas and corrective actions, to set goals, to take strategic decisions and to monitor improvement actions.

3.1 The Indicators System

From the set of indicators, Growela is expecting to obtain:

- improvement measures;
- extra involvement from the staff in obtaining the aimed results;
- identification of procedures that are not producing the required results;
- ease of implementation and adjustment of new management procedures.

Each indicator is described through the following information:

- a description,
- a measurement unit,
- a calculation rule,
- a calculation frequency,
- a level of detail,
- a responsible,
- a data source,
- a target group,
- a current value,
- previous values, and
- a final goal.

These indicators were defined according to a top-down approach, defining levels at which detail and measuring frequency needed to be increased.

The definition of the indicators set was undertaken by a working group specially put together with that purpose.

3.2 Performance Indicators

Growela analyzed their logistic processes, from an high level point of view, and identified their major fields of interest (see **Figure 3-1**):

- commercial department;
- technical design;
- planning;
- purchase department;
- production management;

- quality assurance;
- maintenance control;
- delivery.

Figure 3-1 - Levels of abstraction in GROWELA indicators set. As the level decreases, the calculation frequency increases.

As an example, for quality assurance, the following indicators were chosen:

- inspection reject rates, per product, per section or operation, per supplier dissecting for each raw material, and per product dissecting for each section;
- non conformities, per product, per section, per disposition for a specific product, per non conformity cause, per disposition for a specific supplier and a specific product;
- production return rates, global and per product;
- repair rates, global and per product for each operation;
- scrap rates, global and per product for each operation;
- suppliers returns rates, global, per supplier and per supplier for each product;
- corrective actions, including the number of corrective actions per product and per area, the success rates and the average number of days in retard in the corrective actions effectively implemented.

For the maintenance department, the required indicators were:

- Time analysis, including MTBF, MTTR and MTA, the operators time by intervention, man-power time by intervention and equipment, testing time per equipment and intervention, equipment operational time, man-power yield and lubrication cost (expressed in terms of intervention time for this particular operation);
- Costs analysis, including dissection by intervention, equipment, man-power per equipment, man-power per intervention, materials per equipment and per intervention, spares per equipment and per intervention, total maintenance cost, subcontracts per equipment and per intervention, corrective and preventive maintenance and quality of maintenance actions per equipment.

3.3 Legacy Systems Integration

At Growela, system integration was made at the database level. Growela had a MRP system (Gestical, by Softel) and a LIM system (GCI, also by Softel). However, these systems were built over a standard commercial database server (Informix Online) which was fully supported by all applications.

The data collection system was integrated with a special application which made the interface between the shop-floor terminals and the database.

4. CONCLUSION

The overall conclusion is that a TdB application is useful within the context of new manufacturing management concepts. The prime goal was the construction of an application with the following requirements:

- Usefulness.

Performance indicators are related, at the various management levels, with company departments. Therefore, the TdB must be useful for people working in those areas, both in its semantic content and in the way information is shown to the users.

- Simplicity.

It must be easily accessed by everyone, that is, it should not require deep knowledge on any subject. The information should be shown in such a way that it can be easily perceived and understood.

- Flexibility.

As concepts and strategic goals evolve, so must performance indicators do it also. It should be easy to define new indicators and adapt the existing ones.

- Completeness.

Regarding performance evaluation, the defined set of indicators should be enough to report and document current shop-floor status. This set of indicators must be made available in the TdB application, so that only one interface is presented to the shop-floor.

- MES integration.

Since MES are the primary data source, a very well integrated software solution must be available in order to extract the maximum performance from the system. Nevertheless, the connection requires a deep study on new software development approaches and, specially, in new data sources.

5. BIBLIOGRAPHY

- [1] Beatriz Ribeiro, Paula Silva, João José Ferreira, José Mendonça; "Analysis of PCM tools and standards and specification of PCM tool set"; RIC deliverable D.3.2.1; 95/10/30
- [2] João Neta, Rui Picas, Beatriz Ribeiro, José Mendonça; "New concept of data encapsulation and manufacturing data servers; Server functionality specification according to client application requirements"; RIC deliverable D.4.3.1/2; 95/10/30
- [3] João Neta, Rui Picas, Beatriz Ribeiro, José Mendonça; "Specification of Distributed Heterogeneous System Architecture - Final Specification"; RIC deliverable D.4.1.2; 95/10/30
- [4] Brian Maskell; "Performance Measurement for World Class Manufacturing"; Productivity Press
- [5] João Neta; "Quadros de Bordo Dinâmicos"; Msc paper under the Industrial Automation subject; 95/07/31
- [6] Eve Chapello, Marie-Hélène Delmond; "Les tableau-de-bord, outils de introduction du changement"; Revue Française de Gestion

- [7] "MES to COMMS integration: declaration of interdependenmce"; AMR Report; 92/08/31
- [8] "Cleaning up the MES: positioning manufacturer's needs with software vendors capabilities"; AMR Report; 92/01/31
- [9] Jim Esch; "A fine MES"; BYTE; 95/12/31
- [10] Robert Orfali, Dan Harkey, Jeri Edwards; "Client-Server computing"; BYTE; 95/04/30
- [11] "MES in process manufacturing"; CIME review
- [12] Brian Smith; "Three-tiers for client-server"; Data-based advisory; 96/10/31