Product-Process-System Information Formalization

Marcello Colledani, Walter Terkaj, Tullio Tolio

Abstract This chapter introduces a conceptual framework for the integrated modeling of product, process and production system data. The work focuses on the Manufacturing System Design problem and aims at providing a common data structure as a reference for different methodologies and tools in this domain. The framework is flexible, extendible, scalable and has been developed as an object-oriented model by means of UML (Unified Modeling Language). Moreover, the proposed data model can have a wider applicability since it is based on shared standards and previous general frameworks. The concept of evolution has been introduced into the model, since it is essential to include market uncertainty in the design of competitive production systems. Finally, the developed framework has been translated into a relational database which can be interfaced with all the main phases of the system design approach.

Keywords: Manufacturing Information Formalization; Production System Data; Product Data; Manufacturing Process Data; STEP-NC

4.1 Introduction

Chapter 1 has already introduced the problem of manufacturing system design, highlighting the wideness and complexity of the activities that it is necessary to carry out in order to obtain effective system solutions (see Sect. 1.1). Above all, the turbulence of the market environment makes hard to tackle the configuration, reconfiguration, implementation, management, control and continuous improvement of the production systems which have to cope with the changes of as products and processes.

This chapter aims at establishing a common view and defining a common structure to handle the information used and generated during the design of Manufacturing Systems. From the industrial standpoint, this problem is highly critical since on one side many economical and technological issues must be jointly considered and on the other the final result has a strong impact on the profitability of the firm.. System design can be time consuming and expensive, since qualitative and quantitative aspects are analyzed. Therefore there is the need of support tools to make the procedure of system configuration more efficient, so reducing the process time, and more effective, so increasing the chance to design the best configuration (Cantamessa et al. 2007). Chapter 1 introduced system design as a complex problem which requires to share data among different activities (Fig. 1.3) including manufacturing strategy, process planning, system configuration, capacity planning and performance evaluation. An holistic and integrated view is necessary to model the most important relations among the different aspects of the manufacturing environment as well as it is necessary to develop a unique standard framework to formalize data on products, processes and production systems. All the data necessary for system design activities have been formalized by following an innovative integrated data structure for evolving product-process-system. This integration is fundamental to optimally address the design problem, because most of the data are strongly related. The data framework has been enriched with the evolutionary concept which has been introduced to consider possible changes of the system (i.e. reconfigurations) as well as of the products/processes. Given the uncertain environment manufacturing firms have to cope with, it comes as fundamental to consider the evolutionary dynamic when designing a production system. In fact, the system life-cycle is longer than the product life-cycle and, since the best system configuration is time-varying according to the product evolution, some system reconfigurations could be required to optimally address the production problems.

The data formalization introduced in this chapter has been adopted as a reference by all the methodologies and tools that are described in the chapters of this book. The following section briefly analyzes the literature, Sect. 4.3 profiles a detailed object-oriented data formalization, while Sect. 4.4 presents its implementation in a relational database.

4.2 Literature Analysis

The problem of data formalization has received much attention in literature: knowledge-based analysis methodologies and tools have been developed to support the decision making processes all over the product/process/system life-cycles. Bernard and Tichkiewitch (2008) made a contribution to this area, proposing a book with a complete overview of the knowledge life-cycle management topic with the most recent and innovative results. However, data evolution is not fully addressed by academic models and is not faced at all by industrial standards. Moreover, the integration issues have not yet been solved.

A framework to manage manufacturing information should be able to support the user in the production modeling activity and feed the decision support tools with the required information. Casati and Pernici (2001) have outlined four main requirements for a knowledge management framework:

- Flexibility: the model must be easily adaptable in order to describe many different production system architectures, processes and product features.
- Extendibility: the model must guarantee the potential for the user to rapidly extend the range and/or detail level, if needed.
- Scalability: the model must be able to support product, process and production system descriptions at different levels of detail.
- Integration: products, production processes and systems, together with the relationships among them, must be considered and described in the same framework, since they all belong to the manufacturing environment.

The problem of developing a data formalization framework for a manufacturing context has been traditionally faced by proposing solutions which can be in some cases easily adapted to different situations but which do not take into account all the requirements defined above. The main drawback is that existing models generally consider products, processes and production systems as separated from each other, and integration is not fully addressed. Moreover, the evolutionary aspect has been hardly faced, even if it has a deep impact on the performance of a manufacturing system (see Sect. 1.4).

The following sub-sections briefly analyze the literature about product, process, system and their integration.

4.2.1 Product and process

Product life-cycle issues have been examined in recent years from many different viewpoints. The available models range from marketing models, dealing with curves describing the evolution of product demands, to models dealing with the evaluation of the environmental impact on the product life-cycle (LCA – Life-Cycle Analysis), to cost management models dealing with the costs related to the various phases of the life-cycle, to a more technical approach which considers product-process engineering activities along the life-cycle. What is still missing is a complete view from the perspective of the producer of manufacturing systems. This view should integrate the various viewpoints providing information to support the design of the production system life-cycle connected with the product life-cycle. The completeness attributed to this viewpoint is strictly connected to the opportunity to describe the product evolution over time in terms of number and type of product variants, sales volumes and proper characterization of each variant, in terms of both raw piece description (e.g. shape and material) and technological operations to be performed on it (i.e. process plan). Heterogeneous information must be considered during the modeling phase, given that defining a single language, a single methodology and a single software tool is tough.

The uncertainty affecting the product evolution, nowadays more critical than ever, should also be modeled in the most appropriate way. Important product information models are provided by the industrial standards such as STEP (ISO 10303) and PLCS (ISO 10303-239). Anyway, these standards do not allow to describe both the geometric information about a certain product variant, and the information concerning the process cycle. This results in a static description of a product type, not considering the uncertain evolution over time of the different product variants.

PSL (Process Specification Language) project (ISO/CD18629 2002) is an interesting approach in the field of process knowledge formalization. PSL provides a language for process data exchange to integrate multiple applications handling data related to processes throughout the manufacturing context life-cycle.

Paying attention to the integration between product and process, the STEP-NC standard (ISO 14649) presents a model of data interoperability between CAD/CAM systems and CNC machine tools. This standard has been taken as a reference to develop the product and process side of the data formalization presented in this chapter.

4.2.2 Production System

From the production system standpoint, many works have been developed in the past to define languages and methodologies for their description, analysis and design. All these tools are useful but can be used only to describe the static and dynamic behavior of a single version of the production system. However, in everyday practice a system can undergo many kinds of evolution, driven mainly by product and process modifications. To model such changing systems, tools for the description of different evolution scenarios, with the related information concerning probabilities and durations, should be defined and used.

Many works adopted an object-oriented approach to model manufacturing systems. In these works, the manufacturing system is decomposed into objects instantiated from classes. Each object has an identity, a state and a behavior following the object-oriented paradigm. Van Brussel et al. (1995; 1998) presented a holonic reference architecture for manufacturing systems modeled with UML class diagrams. Park et al. (1997) proposed an object-oriented modeling framework called JR-net for a generic AMS (Automated Manufacturing System); resource-type, job-type and control-type objects compose the model of a generic AMS. Kellert et al. (1997) proposed a conceptual model for FMSs (Flexible Manufacturing Systems). Booch et al. (2004) proposed another object-oriented model for FMSs. The authors adopted the OMT (Object Modeling Technique) formalism to model the static portion of the system, DFD (Data Flow Diagram) for the dynamic and functional models, and STD (State Transition Diagram) for the control aspects.

Bruccoleri et al. (2003) and Matta et al. (2004) proposed UML-based modeling approaches to describe all static and dynamic aspects of respectively a cell controller and a complete FMS.

4.2.3 The integration

The need for integration of the two previous aspects (the product/process and the production system) directly derives from the viewpoint of the machine tool builder who wants the vision on the design and management of his product life-cycle (i.e. the production system seen as a product according to the vision proposed by the Manufuture Platform) to be as complete as possible. Moreover, the machine tool builder aims at deriving a guidance to handle the most critical issues occurring within the system design problem. For example, starting from the description of how the product will probably evolve over time, it is possible to take very critical decisions, e.g. whether to acquire some kind of flexibility at a certain degree, focused on some aspects relevant to his production problem, or to opt for the choice of a more rigid and productivity-oriented system solutions.

Regarding the integration of information, Kimura (1993) proposed a modeling framework for product and process under a virtual manufacturing point of view.

Thibault et al. (2006) presented a tool called "Ontoforge" to support the integrated design of a forged product considering the knowledge about the process and the information about the system. López-Ortega and Moramay (2005) presented a meta-model using Express-G formalism to include STEP standard in a flexible manufacturing domain.

Bernard et al. (2006) proposed a meta-model structure to link the function/behavior/structure applied to either product, process or resources and external effects.

Colledani et al. (2008) have developed a formalized link between the production system side and the product/process side. Their work proposed a conceptual reference framework for the integrated modeling of product, production process and system data. The framework consists of an object-oriented model by means of the UML de-facto standard. The class diagram of this UML model, representing the core of the framework, is described in detail. The conceptual reference framework was developed to support both researchers and industrialists while modeling their problem solving methodologies. The basic idea is that a more effective use of heterogeneous decision support methods at different enterprise levels can be obtained if these methods are based on a common conceptual model. The authors have proposed also two initial applications of the reference framework; one of these applications is about manufacturing system design and represents an earlier proposal of the data formalization presented in Sect. 4.3. The main innovation of the work by Colledani et al. (2008) consists in the integrated representation of products, processes and production systems data, information and knowledge, satisfying all the requirements previously highlighted. This work has partly inspired the data structure presented in this chapter but the latter has been further developed focusing the attention on the manufacturing system side and introducing the concept of evolution.

4.3 Data Formalization for Manufacturing System Design

The system design process plays a key role in defining the overall performance of competitive manufacturing systems having to face the trade-off between productivity and flexibility. The problem consists of designing the optimal system configuration, i.e. the number and type of resources needed to properly satisfy the demand. Technological requirements of the part types to be produced have a major impact on the selection of the typologies of resources to be adopted, while the production volume requirements influence primarily the choice of the number of resources (Tolio and Valente 2006). The system configuration and reconfiguration problem requires a data structure not only to describe an existing system but also to formalize the elements that can be added (removed) to (from) the reconfigured system.

An effective support tool needs a complete and precise data formalization. For this purpose, a reference framework has been developed by adopting the UML Class Diagram formalism (Figure 4.1).

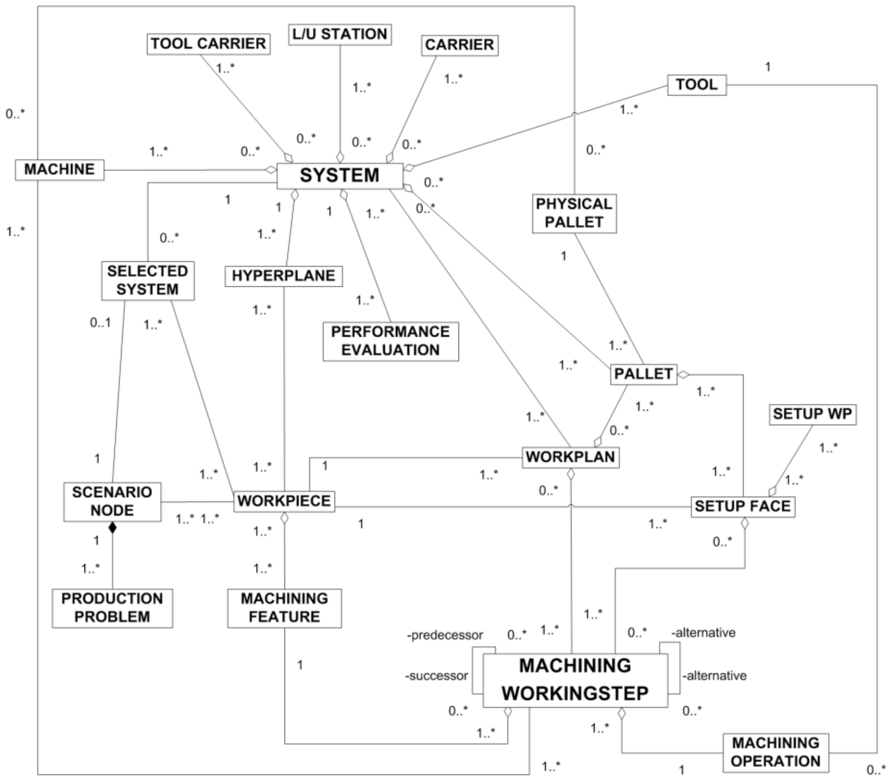

Fig. 4.1 UML Class Diagram

The Class Diagram in Figure 4.1 shows three main areas: Product, Production System and Process.

In the Product area, the technological and demand characteristic of the product are described. The following classes have been defined:

- Workpiece;
- Machining Feature;
- Scenario Node;
- Production Problem.

In the Production System area the resources and the system characteristics are defined. Since in all this book the attention is centered on Focused Flexibility Manufacturing Systems (FFMS) and on Flexible Manufacturing Systems (FMS), the system resources can be machines, carriers, tool carriers, load/unload stations and pallets. Data related to the manufacturing system and its components are detailed by the following classes:

- System;
- Selected System;

- Hyperplane;
- Machine;
- Carrier;
- Load/Unload Station;
- Physical Pallet;
- Tool;
- Tool Carrier;
- Performance Evaluation.

The Process area describes, by means of the following classes, how the production system can produce the products:

- Machining Operation;
- Machining Workingstep;
- Workplan;
- Pallet;
- Setup Face;
- Setup WP.

These three areas have been highlighted in the UML Class Diagram as shown in Figure 4.2.

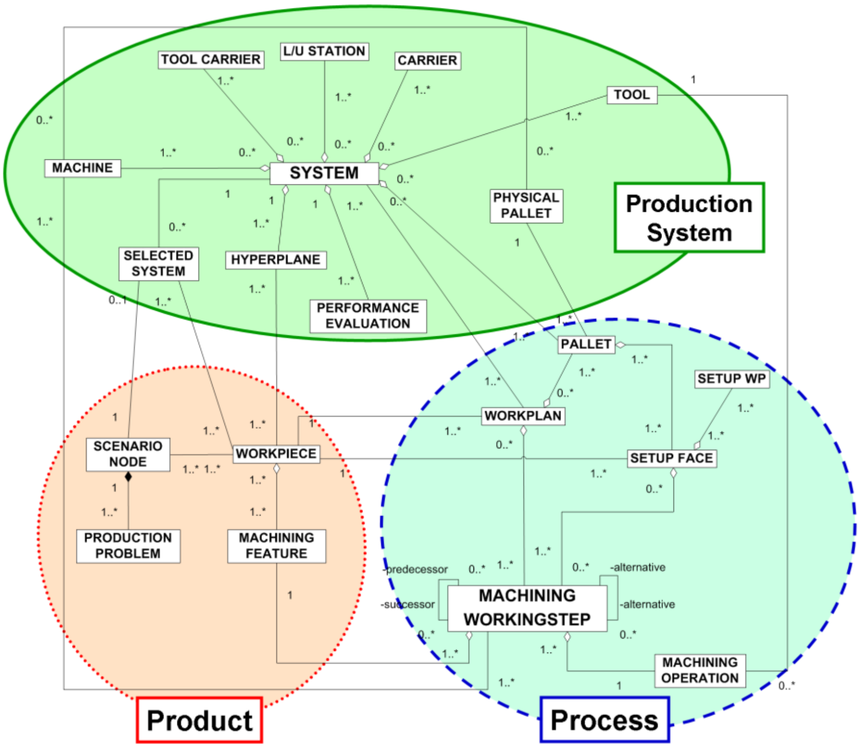

Fig. 4.2 Areas of the UML Diagram

The following sections describe all these classes and their attributes.

4.4 Product

The “Product Area” (red dotted line in Figure 4.2) consists of the Workpiece (Table 4.1; Figure 4.3), Machining Feature (Table 4.2; Figure 4.4), Scenario Node (Table 4.3) and Production Problem (Table 4.4) classes; the first two classes are partially derived from the STEP-NC standard (ISO 14649). Each instance of the Workpiece class is one of the part types produced by the system and is related to the codes which can be ordered by the customers.

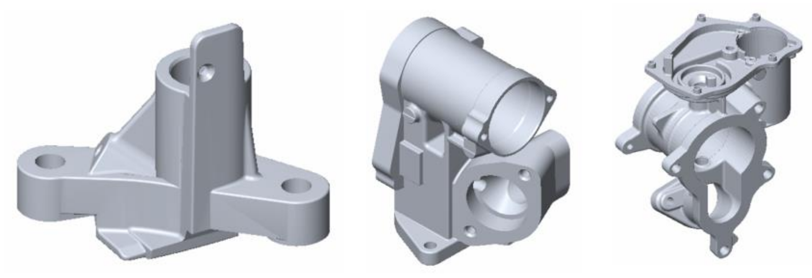

Fig. 4.3 Workpiece Examples

Table 4.1 Workpiece Attributes

Attribute Name	Attribute Definition
id_workpiece	Workpiece identifier
its_material	Workpiece material
global_tolerance	Workpiece general tolerance
its_rawpiece_geometry	Workpiece raw piece geometry
its_geometry	Final workpiece geometry
x_bounding_pos	Positive coordinate along x-axis of bounding geometry
x_bounding_neg	Negative coordinate along x-axis of bounding geometry
y_bounding_pos	Positive coordinate along y-axis of bounding geometry
y_bounding_neg	Negative coordinate along y-axis of bounding geometry
z_bounding_pos	Positive coordinate along z-axis of bounding geometry
z_bounding_neg	Negative coordinate along z-axis of bounding geometry
pen_out_cost	Penalty related to missing production; it can figure as an extra cost if outsourcing is adopted
ru_coeff	Product ramp-up coefficient

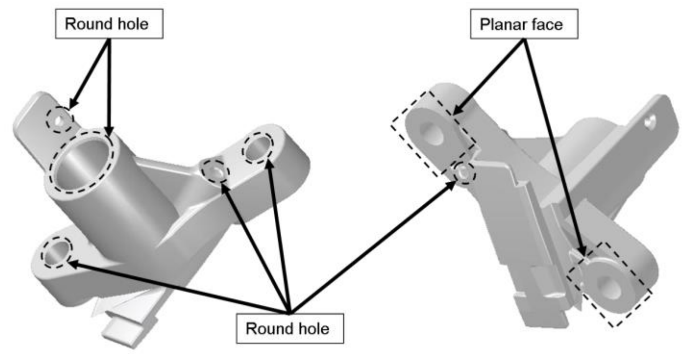

Fig. 4.4 Machining Feature spotting

Each Workpiece is characterized by a set of Machining Features (Table 4.2), i.e. geometric modifications which are realized by machining operations starting from the raw piece. According to ISO14649, there are different types of feature. Among them the most important are: planar face, pocket, slot, step, hole, generic feature and compound feature.

Table 4.2 Machining Feature

Attribute Name	Attribute Definition
id_feature	Identifier
its_workpiece	The workpiece which the feature is part of
its_operations	A set of all (machining) operations associated with the feature required to manufacture the feature. In this set of operations there can be alternative operations; for example, if a feature can be machined on two different machines then it is likely that the machines require a different operation (e.g. different cutting speed, feed, etc.)
abstract_supertype	The type of feature (e.g. planar_face, pocket, slot, step, round_hole, toolpath_feature, profile_feature, boss, spherical_cap, rounded_end, thread)
placement_location_x	Position of the feature along the x-axis in the workpiece coordinate system
placement_location_y	Position of the feature along the y-axis in the workpiece coordinate system
placement_location_z	Position of the feature along the z-axis in the workpiece coordinate system
cos_x	Cosine of the angle between the feature working direction and the x-axis of the workpiece coordinate system
cos_y	Cosine of the angle between the feature working direction and the y-axis of the workpiece coordinate system
cos_z	Cosine of the angle between the feature working direction and the z-axis of the workpiece coordinate system

The workpiece demand is affected by both mid-term and long-term variability. Long-term variability is modeled through the “Scenario Node” class which contains the demand evolutionary data according to a scenario tree representation. Since the production problem resulting from the combination of many products can be pretty hard to manage under an evolutionary perspective, the scenario tree representation (Figure 4.5) has been adopted to simplify the problem formulation (Ahmed 2003). Each scenario node is characterized by a realization probability and keeps a set of production problems inside. In fact, mid-term variability is modeled by the “Production Problem” class (Table 4.4) whose instances are production contexts that a manufacturing system should be in principle able to satisfy without requiring a major reconfiguration. One Scenario Node explodes into one or more Production Problems (Figure 4.5).

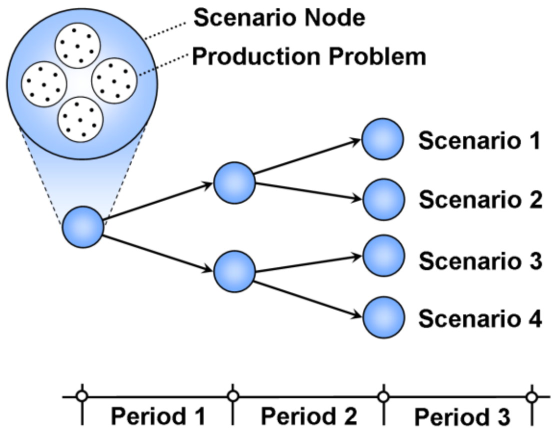

Fig. 4.5 Scenario Tree

Table 4.3 Scenario Node

Attribute Name	Attribute Definition
id_scenario_node	Node identifier
id_parent_node	Parent node identifier
time_stage	Node time stage
time_step	Time period length which the demand is referred to (the length is the same for the nodes of the same time stage)
probability	Realization probability
mean_part_mix_demand	Mean demand volume of the workpieces in the scenario node
min_demand	Minimum demand volume of the workpieces in the scenario node
max_demand	Maximum demand volume of the workpieces in the scenario node
min_agg	Minimum value of the aggregate demand volume in the scenario node
max_agg	Maximum value of the aggregate demand volume in the scenario node
budget	Budget of the system user that is available for investment in a new system configuration
discount_rate	Discount rate associated to the node. It can be seen as a measure of risk as perceived by the system user
sel_mtb	Parameter equal to 1 if the scenario node is considered by the machine tool builder when studying the production problem
sel_user	Parameter equal to 1 if the scenario node is considered by the system user when studying the production problem

Table 4.4 Production Problem

Attribute Name	Attribute Definition
id_prodprob	Production problem identifier
id_scenario_node	Identifier of the scenario node which the production problem belongs to
part_mix_demand	Demand of the workpieces in the production problem

Both Scenario Node class and Production Problem class play a key role in the data model because they transform the framework from "static" to "dynamic", by describing the stochastic data of the problem (see Sect. 7.3 and Sect. 7.4.1).

4.5 Production System

In the "Production System Area" (green continuous line in Figure 4.2), the architectural characteristics of the manufacturing system are detailed. The System class (Table 4.5) is crucial since the definition of the system configuration is the final goal of the whole data formalization process and the system design problem itself. The system configuration is the technical solution proposed by a Machine tool builder; this configuration can modify an existing system or define a new manufacturing system from "green field".

The dynamics of the manufacturing system configuration can be also represented thanks to the "previous_system" attribute, which links a given system with its previous configuration. The link between the system configuration and the addressed production context is represented by the Selected System class (Table 4.6) which holds the decisions taken by the system user about planning system capacity. The Selected System attributes detail which system configuration is chosen to face a particular Scenario Node and how the system configuration performs when facing that context (see Chapter 8).

A System configuration is characterized by the possible combinations of production volumes that the system can yield. This property is modeled by the class Hyperplane (Table 4.7). The instances of Hyperplane represent the hyperplanes which are required to mathematically define the admissible production domain of the system configuration (see Sect. 7.4.4).

The System is composed by its physical resources: Machines (Table 4.8), Carriers (Table 4.9), Load/Unload Stations (Table 4.10), Tools (Table 4.11), Tool Carrier (Table 4.12) and Physical Pallets (Table 4.13). A Physical Pallet is the element consisting of various sub-elements (table, fixture, etc.) on which the workpieces are mounted and that enters the machine to hold the workpieces while being processed (see Sect. 7.4.2).

The instances of the previous classes are either the types of resources composing the current system configuration or the types of resources which are available in the catalogue of the machine tool builder. Their attributes consist of the technological, physical and cost characteristics.

Moreover, the Performance Evaluation class (Table 4.14) has been put into the model to define which system configurations must be evaluated. In this book the performance of the system are evaluated through the simulation technique (see Chapter 9), therefore the attributes of the class have been defined accordingly.

Table 4.5 System

Attribute Name	Attribute Definition
id_system	System configuration identifier
previous_system	Previous system configuration identifier
lead_time	Lead time from the order issue to the installation of the new system configuration
inv_cost	Investment cost of the new system configuration (i.e. the commercial bid of the machine tool builder) [€]
op_cost	Operating cost of the system configuration [€]
machine_N	Number of machines for each machine type in the system configuration
carrier_N	Number of carriers in the system configuration
Lustation_N	Number of L/U stations in the system configuration
pallet_N	Number of pallets for each pallet type in the system configuration
tool_N	Number of tools for each tool type in the system configuration
open_time	Daily opening time of the system [min/day]
workplans	Set of workplans that are processed in the system
ru_coeff	System ramp-up coefficient
sat_conf	Resource type saturation (value estimated by the system configuration activities)
sat_sim	Resource type saturation (value estimated using simulation)
vol_conf	Production volume of the system configuration in a scenario node (value defined by the system configuration activities)
vol_plan	Production volume of the system configuration in a scenario node (value defined by capacity planning activities)
vol_sim	Production volume of the system configuration in a scenario node (value estimated using simulation)
vol_plan_miss	Missing production volume of the system configuration in a scenario node (value estimated by capacity planning activities)
penalty	Penalty related to missing production; it can be an extra cost if outsourcing is adopted [€]

Table 4.6 Selected System

Attribute Name	Attribute Definition
id_system	Selected system configuration identifier
root_node	Identifier of the scenario node which is the root of the considered scenario tree
time_stage	Time stage during which the system configuration must be implemented

Table 4.7 Hyperplane

Attribute Name	Attribute Definition
id_hyperplane	Hyperplane identifier
id_system	Identifier of the system which the hyperplane belongs to
id_resource	Resource type which the hyperlplane is related to
resource_N	Number of resources for each resource type (e.g. machine, carrier, etc.) in each system configuration
rhs	Right hand side, i.e. total capacity of the resource type
hyper_coef	Coefficients of the hyperplanes defining the Admissible Domain of the system configuration
operator	Operator of the constraint: GT = greater than, GE = greater or equal, E = equal, LT = less than, LE = less or equal
work_cost	Production cost for each resource [€]

Table 4.8 Machine

Attribute Name	Attribute Definition
id_machine	Machine type identifier
avail	Daily availability of the machine type [min/day]
investment_cost	Investment cost of the machine type [€]
axis_number	Number of controlled axes in the machine
axis_characteristics	Characteristics of the axes of the machine
dim_x	Dimension x of the machine type [mm]
dim_y	Dimension y of the machine type [mm]
dim_z	Dimension z of the machine type [mm]
wcube_x	Work cube dimension along x-axis [mm]
pos_trav_x	Positive travel along x-axis [mm]
neg_trav_x	Negative travel along x-axis [mm]
speed_x	Speed in rapid movement along x-axis [mm/min]
accel_x	Acceleration in rapid movement along x-axis [mm/s^2]
wcube_y	Work cube dimension along y-axis [mm]
pos_trav_y	Positive travel along y-axis [mm]
neg_trav_y	Negative travel along y-axis [mm]
speed_y	Speed in rapid movement along y-axis [mm/min]
accel_y	Acceleration in rapid movement along y-axis [mm/s^2]
wcube_z	Work cube dimension along z-axis [mm]
pos_trav_z	Positive travel along z-axis [mm]
neg_trav_z	Negative travel along z-axis [mm]
speed_z	Speed in rapid movement along z-axis [mm/min]
accel_z	Acceleration in rapid movement along z-axis [mm/s^2]
pos_trav_B	Positive travel around B-axis [degree]
neg_trav_B	Negative travel around B-axis [degree]
speed_B	Speed in rapid movement around B-axis [round/min]
accel_B	Acceleration in rapid movement around B-axis [$degree/s^2$]
pos_trav_tilting	Positive travel around tilting-axis [degree]
neg_trav_tilting	Negative travel around tilting-axis [degree]
speed_tilting	Speed in rapid movement around tilting-axis [round/min]
accel_tilting	Acceleration in rapid movement around tilting-axis [$round/s^2$]
power	Maximum machine power[kW]
spindle_speed	Maximum spindle speed [rounds/min]
efficiency	Machine efficiency (necessary to calculate usable power)
tool_change_time	Time to change a tool on the machine type [min]
rotation_time	Shuttle rotation time [min]
tool_magazine	Number of slots in the tool magazine
failure_interval_type	Distribution of the time between failures

Attribute Name	Attribute Definition
failure_interval_mean	Mean time between failures
failure_interval_stdev	Standard deviation of time between failure
repair_interval_type	Repair time distribution
repair_interval_mean	Mean repair time
repair_interval_stdev	Standard deviation of repair times
operation_family	Operation family that the machine can execute (0 = prismatic; 1 = rotational)
precision_level	Precision level of the machine (0 = roughing; 1 = finishing)
x_pallet	Position of the origin of the pallet coordinate system along the x-axis of the machine coordinate system
y_pallet	Position of the origin of the pallet coordinate system along the y-axis of the machine coordinate system
z_pallet	Position of the origin of the pallet coordinate system along the z-axis of the machine coordinate system
pallet_dim	Dimension of the pallet table which can be loaded on the machine

Table 4.9 Carrier

Attribute Name	Attribute Definition
id_carrier	Carrier identifier
investment_cost	Investment cost a unit of carrier [€]
avail	Carrier daily availability [min/day]
speed_carrier	Carrier speed [m/min]
LU_time_carrier	Time to load/unload a pallet from the carrier [min]
mean_trasp_time	Mean transportation time for one carrier mission [min]

Table 4.10 Load/Unload Station

Attribute Name	Attribute Definition
id_LUstation	Load/unload Station identifier
investment_cost	Investment cost for a unit of Load/unload station [€]
avail	Daily availability of the load/unload station [min/day]
operators	Number of operators working on the load/unload station
buffer	Number of slots in the load/unload station buffer
pallet_dim_min	Minimum dimension of the pallet that can be loaded
pallet_dim_max	Maximum dimension of the pallet that can be loaded

Table 4.11 Tool

Attribute Name	Attribute Definition
id_tool	Tool type identifier
life	Tool life [min]
regeneration_time	Regeneration time [min]
diameter	Tool diameter [mm]
length	Length of the cutting edge [mm]
tool_length	Length of the tool [mm]

Table 4.12 Tool Carrier

Attribute Name	Attribute Definition
id_tool_carrier	Tool carrier identifier
speed_tool	Tool carrier speed
LU_time_tool	Load/Unload time
LU_time_tool_central	Load/Unload time from central magazine
investment_cost	Investment cost for each unit [€]

Table 4.13 Physical Pallet

Attribute Name	Attribute Definition
id_physical_pallet	Physical pallet identifier
dim_x	Table dimension along x-axis [mm]
dim_y	Table dimension along y-axis [mm]
dim_z	Table dimension along z-axis [mm]
fix_dim_x	Fixture dimension along x-axis [mm]
fix_dim_y	Fixture dimension along y-axis [mm]
fix_dim_z	Fixture dimension along z-axis [mm]

Table 4.14 Performance Evaluation

Attribute Name	Attribute Definition
id_simulation	Simulation run identifier
id_system	Identifier of the system to be evaluated through simulation
id_scenario_node	Identifier of the scenario node to be evaluated through simulation
replicates	Number of replicates
length	Length of the run [min]
warmup	Length of the warm-up period [min]
log	Parameter that is equal 0 if log file must be created; 1 otherwise

4.6 Process

The “Process Area” (blue dashed line in Figure 4.2) describes how the system resources can be used to machine the workpieces. The Machining Operation (Table 4.15), Machining Workingstep (Table 4.16) and Workplan (Table 4.17) classes are partially derived from STEP-NC standard.

Instances of the Machining Operation class describe the machining processes, specifying the tool to be used and a set of technological parameters. Instances of the Machining Workingstep class represent the machining process for a specific machining feature; a machining workingstep defines the association between a distinct feature and an operation to be performed on the feature. As the related operation, the machining workingstep is characterized by the use of a single tool and a set of technological parameters which are usually constant during the application of the machining workingstep. During the machining workingstep, no tool change is allowed.

Table 4.15 Machining Operation

Attribute Name	Attribute Definition
id_operation	Machining operation identifier
retract_plane	The height of a retract plane associated with the operation
its_tool	Identifier of the tool that must be used for this operation
feederate	Feedrate of the tool. The feed rate specified applies to the motion of the tool center point
cutspeed	Cutting speed
coolant	Coolant options
spindle_speed	Required spindle speed [rounds/min]
power	Required power [kW]
operation_type	Operation type (e.g. milling, drilling, turning, etc.)
operation_family	Operation family which the operation belongs to (0 = prismatic; 1 = rotational)
precision_level	Precision level of the machining operation (0 = roughing; 1 = finishing)

Table 4.16 Machining Workingstep

Attribute Name	Attribute Definition
id_workingstep	Machining workingstep identifier
its_feature	The manufacturing feature upon which the machining workingstep operates
its_operation	The operation which will be performed upon the machining feature
its_effect	The change to the geometry of the workpiece caused by the operation. A CAM system can use this attribute to describe the predicted effect of this operation on the geometry of the workpiece
its_secplane	The security plane for the machining workingstep. On or above this plane, i.e. for z-value greater than this, a safe movement of the tool without danger of collision is possible
ws_cutting_time	Cutting time of the machining workingstep
its_tool_direction	Tool direction
machine_set	Set of machines where the machining workingstep can be processed
alternative_ws	Set of alternative machining workingsteps
predecessor	Set of machining workingsteps that are predecessors of the described machining workingstep
together	Set of machining workingsteps that must be processed together (i.e. on the same pallet and same machine) with the described machining workingsteps

Compared to the STEP-NC approach, the Machining Workingstep class shows also the “ws_cutting_time” attribute and the “machine_set” attribute. The “ws_cutting_time” attribute defines the machining time needed to carry out the workingstep, while the “machine_set” attribute represents the set of machine types which can process the machining workingstep.

Process constraints are defined among the instances of the machining workingstep class.

A Workplan (Table 4.17) is defined as collection of Machining Workingsteps together with an execution sequence. Moreover, a Workplan can be seen also as an ordered sequence of Pallet types. A Pallet type (Table 4.18) is the logical element that defines how a Physical Pallet (Table 4.13) can be used to process the workpieces. An example of pallet is shown in Figure 4.6. Each Pallet is composed by a set of faces where the workpieces can be clamped on. The Setup Face class (Table 4.19) defines which is the setup (i.e. the orientation) of the workpieces, while Setup WP class (Table 4.20) defines the locations of the workpieces on the face.

Table 4.17 Workplan

Attribute Name	Attribute Definition
id_workplan	Workplan identifier
id_workpiece	Workpiece processed with the described workplan
workingsteps	Set of machining workingsteps needed to complete the workplan
pallets	Set of pallets with an execution sequence needed to complete the workplan

Table 4.18 Pallet

Attribute Name	Attribute Definition
id_pallet	Pallet identifier
LU_time_type	Distribution of the time to load/unload all the parts on/from a pallet
LU_time_mean	Mean time to load/unload all the parts on/from a pallet
LU_time_stdev	Standard deviation of the load/unload time
physical_pallet	Physical pallet related to the pallet type
N_parts	Number of parts mounted on the pallet
N_setupface	Number of faces on the pallet
machine_set	Set of machines where the pallet can be loaded
x_setupface	Origin of the setup faces along the x-axis in the pallet coordinate system
y_setupface	Origin of the setup faces along the y-axis in the pallet coordinate system
z_setupface	Origin of the setup faces along the z-axis in the pallet coordinate system

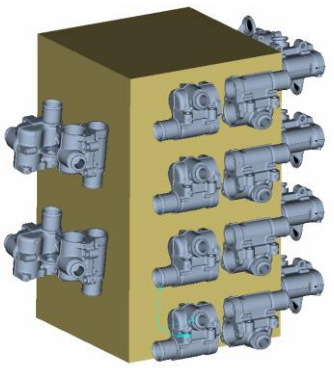

Fig. 4.6 An example of Pallet with four faces

Table 4.19 Setup Face

Attribute Name	Attribute Definition
id_setup_face	Setup Face identifier
pallet	Set of Pallet types where the setup face is present
N_parts	Number of parts that are mounted on the setup face
workpiece	The workpiece type which is processed on the setup face. It is assumed that a setup face can have only one workpiece type (see Sect. 6.3)
workingsteps	Set of machining workingsteps machined on the setup face
rapid_time	Total rapid time to process a machining workingstep on all the workpieces which are mounted on a setup face. Rapid time depends on the machine where the machining workingstep is executed
cos_xx	Cosine of the angle between the x-axis of the workpiece coordinate system and the x-axis of the machine coordinate system
cos_xy	Cosine of the angle between the x-axis of the workpiece coordinate system and the y-axis of the machine coordinate system
cos_xz	Cosine of the angle between the x-axis of the workpiece coordinate system and the z-axis of the machine coordinate system
cos_yx	Cosine of the angle between the y-axis of the workpiece coordinate system and the x-axis of the machine coordinate system
cos_yy	Cosine of the angle between the y-axis of the workpiece coordinate system and the y-axis of the machine coordinate system
cos_yz	Cosine of the angle between the y-axis of the workpiece coordinate system and the z-axis of the machine coordinate system
cos_zx	Cosine of the angle between the z-axis of the workpiece coordinate system and the x-axis of the machine coordinate system
cos_zy	Cosine of the angle between the z-axis of the workpiece coordinate system and the y-axis of the machine coordinate system
cos_zz	Cosine of the angle between the z-axis of the workpiece coordinate system and the z-axis of the machine coordinate system

Table 4.20 Setup WP

Attribute Name	Attribute Definition
id_setup	Setup WP identifier
workpiece	The workpiece which the setup wp is related to
x_setupwp	X coordinate of the setup wp in the setupface coordinate system
y_setupwp	Ycoordinate of the setup wp in the setupface coordinate system
z_setupwp	Z coordinate of the setup wp in the setupface coordinate system
setupface	Setup face which the setup wp is related to
n_row	Number of rows on the setup face
n_col	Number of columns on the setup face
d_row	Distance between two workpieces on the same row
d_col	Distance between two workpieces on the same column

4.7 Implementation

Implementing the data formalization model described above is a key aspect to be considered, since the proposed framework aims at real industrial world applications. This kind of knowledge can be represented through both a relational database and an ontology. A database can give a more concrete and specific vision of the world, while an ontology is used to create a conceptual model of the world; a database focuses on the instances, while an ontology on the entities. Moreover, an ontology can be analyzed by "reasoning" methods which can help to extend the knowledge.

During the work, it was decided to adopt a relational database implementation in order to guarantee an easier integration and data exchange among the modules composing the system design architecture. The relational database has been implemented using MS Access. An abstract of tables and relations of the database is reported in Figure 4.7.

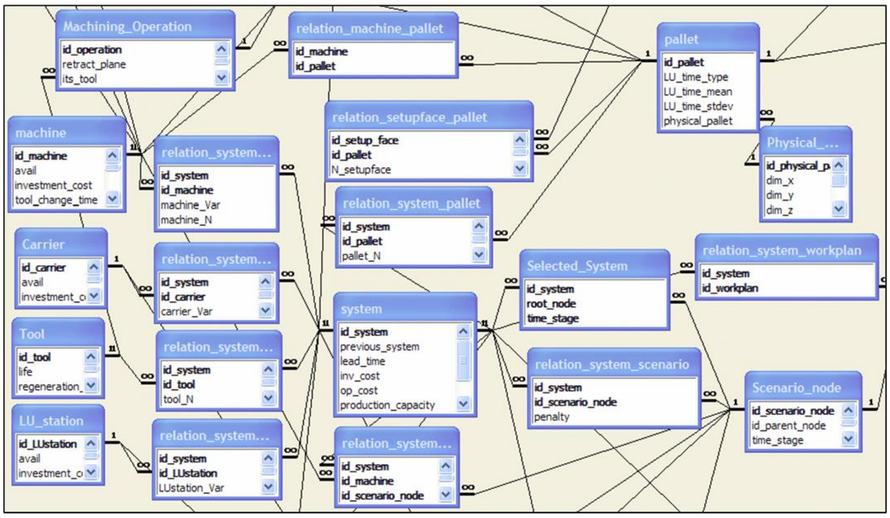

Fig. 4.7 Database Relations

The link between the data formalization classes and the database is shown by the following tables: product data (Table 4.21), resource data (Table 4.22), process data (Table 4.23), link between machine and process (Table 4.24), resources in the system configuration (Table 4.25), process in the system configuration (Table 4.26), performance of the system configuration (Table 4.27) and capacity plan (Table 4.28). It can be noticed that implementing a relational database requires a large number of relations to be made explicit.

Table 4.21 Product Data

Database Table	Related Classes
Workpiece	Workpiece
Machining_Feature	Machining Feature
Scenario_node	Scenario Node
Production_Problem	Production Problem
relation_feature_workpiece	Machining Feature, Workpiece
relation_workpiece_scenario	Workpiece, Scenario Node
relation_workpiece_problem	Workpiece, Production Problem

Table 4.22 Resource Data

Database Table	Related Classes
Machine	Machine
Carrier	Carrier
LU_station	Load/Unload Station
Tool	Tool
Tool_carrier	Tool Carrier
Physical_Pallet	Physical Pallet

Table 4.23 Process Data

Database Table	Related Classes
Machining_Operation	Machining Operation
Machining_Workingstep	Machining Workingstep
Pallet	Pallet, Physical Pallet
Setup_Face	Setup Face
Setup_WP	Setup WP
Workplan	Workplan
relation_feature_operation	Machining Feature, Machining Operation
relation_workplan_workingstep	Workplan, Machining Workingstep
relation_workplan_pallet	Workplan, Pallet
relation_workpiece_pallet	Workpiece, Pallet
relation_setupface_workingstep	Setup Face, Machining Workingstep
relation_setupface_setupWP	Setup Face, Setup WP
relation_setupface_pallet	Setup Face, Pallet
relation_workingstep_workingstep	Machining Workingstep
relation_workingstep_together	Machining Workingstep
relation_workingstep_predecessor	Machining Workingstep

Table 4.24 Link between Machine and Process

Database Table	Related Classes
relation_machine_setupface	Machine, Setup Face
relation_machine_pallet	Machine, Pallet
relation_machine_workingstep	Machine, Machining Workingstep
relation_machine_workingstep_pallet	Machine, Machining Workingstep, Pallet

Table 4.25 System Configuration: Resource Data

Database Table	Related Classes
System	System
relation_system_machine	System, Machine
relation_system_LUstation	System, Load/Unload Station
relation_system_carrier	System, Carrier
relation_system_pallet	System, Pallet
relation_system_tool	System, Tool
relation_system_toolcarrier	System, Tool Carrier

Table 4.26 System Configuration: Process Data

Database Table	Related Classes
relation_system_workplan	System, Workplan
relation_system_machine_pallet	System, Machine, Pallet
relation_system_machine_workingstep	System, Machine, Machining Workingstep
relation_system_machine_workingstep_pallet	System, Machine, Machining Workingstep, Pallet

Table 4.27 System Configuration: Performance Data

Database Table	Related Classes
relation_system_carrier_scenario	System, Carrier, Scenario Node
relation_system_LUstation_scenario	System, Load/Unload Station, Scenario Node
relation_system_machine_scenario	System, Machine, Scenario Node
relation_system_pallet_scenario	System, Pallet, Scenario Node
relation_system_workpiece_scenario	System, Workpiece, Scenario Node
relation_system_scenario	System, Scenario Node
Simulation	Performance Evaluation, System, Scenario Node

Table 4.28 System Configuration: Capacity Plan

Database Table	Related Classes
Selected_System	Selected System
Hyperplane	Hyperplane
relation_workpiece_hyperplane	Workpiece, Hyperplane

4.8 Conclusions

The development of a common data structure for all the activities related to the design of manufacturing systems offers various benefits because it allows the integration of modules, tackling the different sub-problems, that are strongly linked by data exchange. The concept of evolution has been stressed since this kind of information must be provided to methodologies aiming at planning the life-cycle of a manufacturing system in an uncertain environment.

Even if the proposed data formalization has been developed to face the FMS and FFMS design problem, the work can be easily extended to other manufacturing domains thanks to its flexibility and scalability. In particular, information about demand volumes could be further detailed and aspects closer to production planning and system management could be modeled as well.

Other potential future developments could aim at the creation of an object-oriented database to directly implement the data formalization framework described in this chapter without passing through a relational database. For example, an ontology about manufacturing could be realized following the object-oriented model.

Acknowledgments The research has been partially funded by VRL-KCiP (Virtual Research Lab for a Knowledge Community in Production), a Network of Excellence in the 6th Framework Program of the European Commission (FP6-507487-2).

References

Ahmed S, King AJ, Parija G (2003) A Multi-Stage Stochastic Integer Programming Approach for Capacity Expansion under Uncertainty. J Glob Optim 26:3-24

Bernard A, Tichkiewitch S (2008) Methods and Tools for Effective Knowledge Life-Cycle-Management. Springer

Bernard A, Labrousse M, Perry N (2006), LC universal model for the enterprise information system structure. In: Brissaud D, Tichkiewitch S, Zwolinski P (eds) Innovation in Life Cycle Engineering and Sustainable Development. Springer, pp 429-446

Booch G, Jacobson I, Rumbaugh J (2004) The Unified Modelling Language Reference Manual, Second Edition. The Addison-Wesley Object Technology Series

Bruccoleri M, Noto La Diega S, Perrone G (2003) An Object-Oriented approach for flexible manufacturing system control system analysis and design using unified modeling language. Int J Flex Manuf Syst 15:195-216

Cantamessa M, Fichera S, Grieco A, Perrone G, Tolio T (2007) Methodologies and tools to design production systems with focused flexibility. Proceedings of the 4th International Conference on Digital Enterprise Technology, Bath, UK, 19-21 September, pp 627-636

Casati F, Pernici B (2001) Linguaggi per la modellazione dei processi aziendali. In: Batini C, Pernici B, Santucci G (eds) Sistemi informativi. Vol. II: Modelli e progettazione. Franco Angeli

Colledani M, Terkaj W, Tolio T, Tomasella M (2008) Development of a Conceptual Refer-ence Framework to manage manufacturing knowledge related to Products, Processes and Production Systems. In: Bernard A, Tichkiewitch S (2008) Methods and Tools for Effective Knowledge Life-Cycle-Management. Springer

Kellert P, Tchernev N, Force C (1997) Object-oriented methodology for FMS modeling and simulation. Int J Comput Integr Manuf 10:404-434

Kimura F (1993) Product and Process Modelling as a Kernel for Virtual Manufacturing Environment. Ann CIRP 42(1):147-150

López-Ortega O, Moramay R (2005) A STEP-based manufacturing information system to share flexible manufacturing resources data. J Intell Manuf 16:287-301

Manufuture Platform, www.manufuture.org

Matta A, Tolio T, Tomasella M, Zanchi P (2004) A detailed UML model for general flexi-ble manufacturing systems. Proceedings of the 4th CIRP International Seminar on Intel-ligent Computation in Manufacturing Engineering CIRP ICME '04. June 30- July 2, 2004, Sorrento, Italy, pp 113-118

Park TY, Han KH, Choi BK (1997) An object-oriented modelling framework for automated manufacturing system. Int J Comput Integr Manuf 10(5):324-334

Thibault A, Siadat A, Bigot R, Martin P (2006) Method for Integrated Design Using a Knowledge Formalization. Proceedings of 3rd Conference on Digital Enterprise Technology, Setubal, Portugal, 18-20 September 2006

Tolio T, Valente A (2006) An Approach to Design the Flexibility Degree in Flexible Manufacturing System". Proceedings of Flexible Automation & Intelligent Manufacturing Conference, Limerick, Ireland, June 25-272006, pp 1229-1236

Van Brussel H, Valckenaers P, Bongaerts L, Wyns J (1995) Architectural and System De-sign issues in Holonic Manufacturing Systems. Pre-prints of IMS'95, IFAC workshop, Bucharest, Romania, October 1995

Van Brussel H, Wyns J, Valckenaers P, Bongaerts L, Peeters P (1998) Reference Architecture for Holonic Manufacturing Systems: PROSA. Computers in Industry, special issue on Intelligent Manufacturing Systems, 37:255-276