Chapter 2
Holonic Manufacturing Systems

It is well known that manufacturing systems are complex, large-scale systems for a number of operational and structural reasons. This complexity makes such systems difficult to control and predict. Moreover, in order to meet the challenges of "new manufacturing" these systems will need to satisfy fundamental requirements such as [1, 4]: enterprise integration, distributed organization, heterogeneous environments, interoperability, open and dynamic structure, cooperation, integration of humans with software and hardware, agility, scalability, and fault tolerance.

Many manufacturing paradigms promise to meet these challenges. Two of these paradigms, namely, distributed intelligent manufacturing systems and holonic manufacturing systems, have recently been receiving a lot of attention in academia and industry.

Techniques from artificial intelligence have already been used in intelligent manufacturing for more than twenty years [1]. However, recent developments in multi-agent systems in the domain of distributed artificial intelligence have brought about new and interesting possibilities. Distributed intelligent manufaturing systems, or agent-based manufacturing systems, are based on multi-agent system (MAS) technology [5]. MAS studies the coordination of intelligent behavior among a group of (possibly pre-existing) agents. An agent is an autonomous and flexible computational system, which is able to act in an environment [5]. Today, MAS is a very active area of research and is beginning to see commercial and industrial applications.

Over the last twenty years, researchers have been applying agent technology to areas such as manufacturing enterprise integration and supply chain management, manufacturing planning, scheduling and execution control, materials handling and inventory management, to name a few. For an extensive literature review of these applications see [1]. The mainstream applications of agent-based technology for manufacturing systems uses heterarchical architectures as the control paradigm. A heterarchical control system is a flat structure composed of independent entities (agents). These agents typically represent resources and/or tasks. Allocation of tasks to resources is done using dynamic market mechanisms. This yields a simple and fault-tolerant system, since none of the agents need *a-priori* information about the

other agents. As a consequence, several disturbances and changes can easily be coped with. Nevertheless, the basic assumption of this architecture paradigm gives rise to its principle drawbacks that impede the widespread use of these kinds of control systems in industrial environments: the independence of agents prohibits the use of global information. Therefore, global system performance is very sensitive to the definition of the market rules; the control system can not guarantee a minimum performance level in the case of unforseen circumstances; prediction of the behavior of individual orders is impossible, etc. In order to tackle this problem the Intelligent Manufacturing Systems (IMS) consortium[1] initiated a few projects in the early 1990s to define new paradigms for the factory of the "future". The holonic manufacturing system was one of these sets of projects.

In the following sections the holonic background is presented. The state-of-the-art in the field of holonic manufacturing systems is also studied.

2.1 Holon

The holonic concept was developed by the philosopher Arthur Koestler [6] in order to explain the evolution of biological and social systems. On the one hand, these systems develop stable intermediate forms during evolution that are self-reliant. On the other hand, in living and organizational systems it is difficult to distinguish between "wholes" and "parts": almost everything is both a whole and a part, at the same time. These observations led Koestler to propose the word "*holon*", which is a combination of the Greek word "holos", meaning whole, and the Greek suffix "on", meaning particle or part, as in proton or neutron. Koestler observed that, in living organisms and in social organizations, which are entirely self-supporting, noninteracting entities did not exist. Every identifiable unit of organization, such as a single cell in an animal or family unit in a society, comprises more basic units (plasma and nucleus, parents and siblings), while at the same time forming a part of a larger unit of organization (a muscle tissue or a community). The strength of holonic organization, or *holarchy*, is that it enables the construction of very complex systems that are nonetheless efficient in the use of resources, highly resilient to disturbance (both internal and external), and adaptable to changes in the environment in which they exist. Within a holarchy, holons may dynamically create and change hierarchies. Moreover, holons may participate in multiple hierarchies simultaneously. Holarchies are recursive in the sense that a holon may itself be an entire holarchy that acts as an autonomous and cooperative unit in the first holarchy.

The stability of holons and holarchies stems from holons being self-reliant units, which have a degree of independence and handle circumstances and problems on their particular level of existence without asking higher-level holons for assistance. Holons can also receive instructions from and, to a certain extent, be controlled by higher-level holons. This self-reliant characteristic ensures that holons are stable

---

[1] http://www.ims.org/

and able to survive disturbance. The subordination to higher-level holons ensures the effective operation of the larger whole.

2.2 Holonic Manufacturing Systems – HMS

The application of holonic concepts to manufacturing was initially motivated by the inability of existing manufacturing systems (i) to deal with the evolution of products within an existing production facility and (ii) to maintain satisfactory performance levels outside normal operating conditions [4]. Suda introduced the concept of holonic manufacturing in the early 1990s [7] to address the challenge for manufacturing in the 21st century.

Teams of industry experts, scientists, and engineers from the world's leading industrial nations worked together from 1992 to 1994 to build and test a framework for international collaboration in intelligent manufacturing systems (IMS). The experiences of teams coming together from Australia, Canada, Europe, Japan and the USA to work for one year on collaborative "test case" projects formed part of a two-year feasibility study that began in February 1992. This feasibility study proved that this kind of international collaboration could achieve significant results in a relatively short time.

A holonic manufacturing system is based on the concept of "*holonic systems*", developed by Arthur Koestler [6]. Holons in a holonic manufacturing systems assist the operator in controlling the system: holons autonomously select appropriate parameter settings, find their own strategies and build their own structure.

Koestler also points out that holons are autonomous self-reliant units, which have a degree of independence and handle contingencies without asking higher authorities for instructions. Simultaneously, holons are subject to control from (multiple) higher authorities. The first property ensures that holons are stable forms, which survive disturbances. The latter property signifies that they are intermediate forms, which provide the proper functionality for the greater whole. Finally, Koestler defines a holarchy as a hierarchy of self-regulating holons that function (a) as autonomous wholes in supraordination to their parts, (b) as dependent parts in subordination to controls on higher levels, (c) in coordination with their local environment.

Work in the HMS program has translated these concepts to the manufacturing world, viewing the manufacturing system as one consisting of autonomous modules (holons) with distributed control. The goal is to attain the benefits that holonic organization provides to living organisms and societies, in manufacturing, i.e., stability in the face of disturbances, adaptability and flexibility in the face of change, and efficient use of available resources. The HMS concept combines the best features of hierarchical and heterarchical organization [8]. It preserves the stability of hierarchy while providing the dynamic flexibility of heterarchy.

The HMS consortium developed the following list of definitions to help understand and guide the translation of holonic concepts into a manufacturing setting [4]:

- Holon: An autonomous and cooperative building block of a manufacturing system for transforming, transporting, storing and/or validating information and physical objects. The holon consists of an information processing part and often a physical processing part. Figure 2.1 shows the holon general architecture widely used in the field [9]. A holon can be part of another holon.
- Autonomy: The capability of an entity to create and control the execution of its own plans and/or strategies.
- Cooperation: A process whereby a set of entities develops mutually acceptable plans and executes those plans.
- Holarchy: A system of holons that can cooperate to achieve a goal or objective. The holarchy defines the basic rules for the cooperation of the holons and thereby limits their autonomy.
- Holonic manufacturing system: a holarchy that integrates the entire range of manufacturing activities from order booking through design, production, and marketing to realize the agile manufacturing enterprise.

2.3 HMS State-of-the-Art

In the last ten years, an increasing amount of research has been devoted to HMS over a broad range of both theoretical issues and industrial applications. In this section we summarize the main developments reported on specialist HMS literature covering: holon architecture, holons interconnection, holons operation, algorithms for holonic control and methodologies for HMS development.

2.3.1 Holon Architecture

A manufacturing control system for production processes is composed of software modules as well as different physical elements of the manufacturing environment: resources, products, client work orders, coordination operations, etc. The software module and the physical entity, bonded by means of an appropriate communication network, represent a holon in a manufacturing system. Every such holon will be able to reason, make decisions, and communicate interactively with other holons. The number and types of software modules, and the way this software part and the physical entities are interconnected, define the different holon architectures approaches.

The first holon general architecture was proposed by Christensen in 1994 [10]. Figure 2.1 shows the two main components of this architecture: *physical processing part* and *information processing part*. The *physical processing part* is optional. Some examples of holons without a physical processing part are work-order holons, planning holons, scheduler holons, etc. The *physical processing part* is divided into: the physical processing itself, that is the hardware that executes the manufacturing
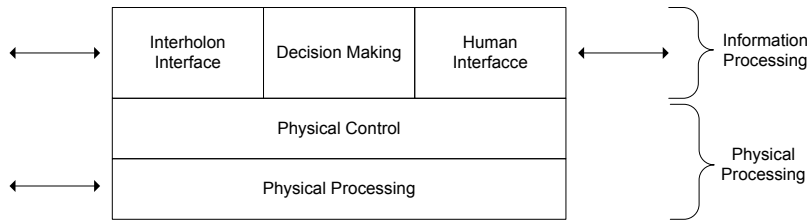
Fig. 2.1 Holon general architecture

operation; and the physical control, a controller (NC, CNC, DNC, PLC) that controls the hardware operation. The *information processing part* is made up of three modules: the holon's kernel or decision making, which is in charge of the holon's reasoning capabilities and decision making; the interholon interface, for the communication and interaction with other holons, and; the human interface, for input (operation commands) and output (state monitoring) data for humans.

An agent-based architecture for the information processing part of Christensen general architecture is proposed in [11]. This proposal is based on the holonic vision of autonomous and cooperation entities. Three main aspects guide this approach. Firstly, holons are entities with autonomous control over the machine behavior they are associated with. Holons may create and execute their own plans and follow their own strategies. This autonomous behavior implies some kind of decision-making component that guides the holon physical control. Secondly, two or more holons are able to cooperate when and wherever it is necessary. To do this, these holons are able to figure out cooperation opportunities, make cooperation or negotiation commitments, and finally to execute the cooperation committed to. Thirdly, holons are able to act in multiple organizations called holarchies and these holarchies are created and modified dynamically. Creating a holarchy means to aggregate the manufacturing process or the controlling process in order to enhance productivity. This implies work and responsibility distribution, and the definition of interaction patterns, which means that holons are able to figure out opportunities for reorganization, negotiate reorganization, and follow the interaction patterns.

The inclusion of these components into the general architecture of Christensen led the authors to propose the agent-oriented architecture in Fig. 2.2. In order to figure out physical behavior and taking into account the current situation, the holon chooses the appropriate plans and strategies in order to reach its long-terms goals. These plans and strategies are communicated from the *decision-making* module to the *behavior control* in order to translate them into hardware operations. On the other hand, the cooperation interactions are initiated by the *decision-making* module and executed by the specific *cooperation techniques* using the *communication techniques* (domain ontologies and languages). In order to reorganize the manufacturing controlling processes the holon needs techniques. These techniques are used to monitor other component actions and to analyze the controlling process. In this way, the holons can figure out opportunities for improvement and can start a negoti-
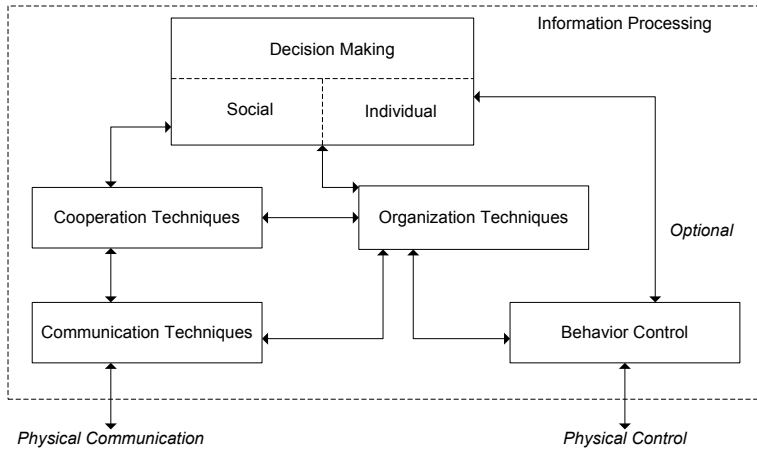
Fig. 2.2 Holon agent-based architecture

ation process for reorganization (*organization techniques*) that is executed by means of cooperation standards.

The Keele University HMS research group, proposes a holon architecture using agents and function blocks [12, 13]. A manufacturing holon is usually composed of knowledge and software modules, as well as an optional hardware component. In terms of functionality, a holon may be considered a composition of an intelligent controlling system (*head*) and a processing system (*base*), Fig. 2.3. The *head* of the holon is based on an agent architecture made up of modules defined in [10]. The elements of the intelligent controlling system are: the *PMC* (process/machine control) executes controlling plans for the running processes; the *PMI* (process/machine interface) provides the logic and physical interface for the processing system through a communication net; the *HI* (human interface) provides the human-readable interface; the *IHI* (interholon interface) is in charge of interholonic communication. The processing system incorporates all the processing modules needed to carry out the production activities. In this way, the *ICS* lets the holon supply the production facilities as autonomous subsystems in coordination with the environment and with other holons. The processing system is responsible for the manufacturing functions defined by the operation rules and strategies imposed by the *ICS*.

In the agent and function block (IEC 61499 [14, 15]) integrated architecture, agents are used to manage high-level planning strategies (*ICS-Head*), while function blocks manage real-time process/machine low-level control (*Base*). There is a holonic kernel running over the function blocks in order to provide the necessary interface between the agents and the IEC 61499. The holarchies and holons' interactions based on agents are organized by means of a structure called a cooperation domain, see Fig. 2.4. A cooperation domain (*composed holon*) is a logic space through which: (i) holons communicate and cooperate, and; (ii) an environment in which holons can find, contact and interact with other holons is provided.

Fig. 2.3 Holon architecture
based on agents and function
blocks
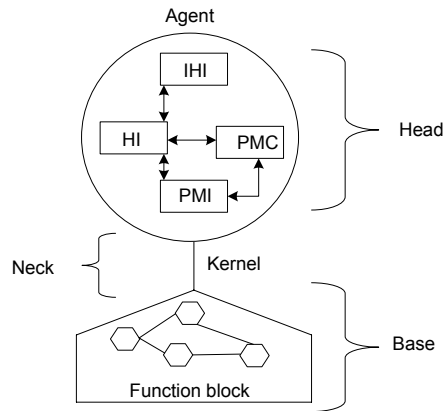
A cooperation domain is not feasible on its own but there must at least be a member holon. Cooperation domains are dynamically created by the execution of the holons' functional components. A holonic system is composed of at least one cooperation domain. A holon can simultaneously be a member of one or more cooperation domains. Every domain is led by a coordinator that is the interface with the outside (other domains). A holon can join a cooperation domain, query its attributes, interchange information with other holons, and exit the domain whenever the holon has finished its task. A cooperation protocol is executed in order to assign tasks to holons in the cooperation domain.

Deen and Fletcher propose a computational model for tasks redistribution (holarchy reorganization). The model is based on *temperature equilibrium* concepts [16]. When a delayed task is processed, the holons may experience overloads and
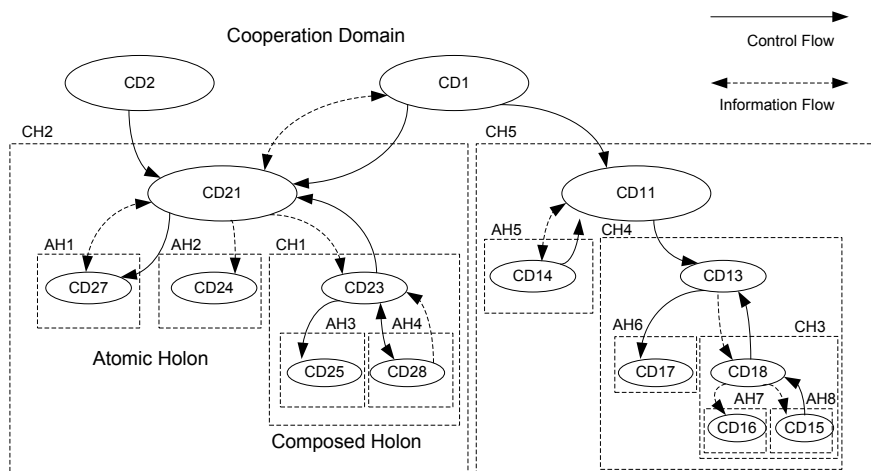

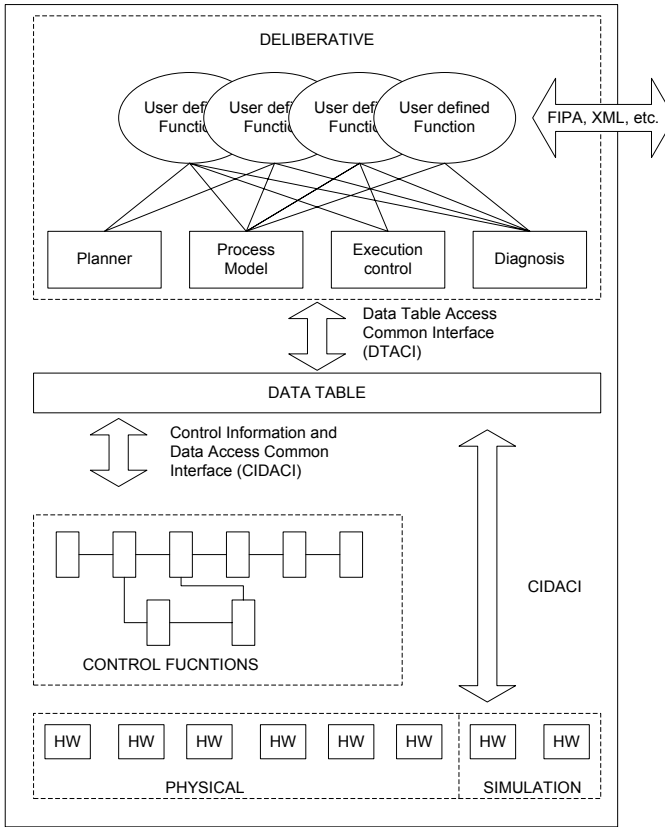**Fig. 2.4** A cooperation domain of the agent and function block integrated architecture

Fig. 2.5 The HCD architecture

this makes the holon "hot". In this way, when a holon realizes that its temperature is over a predefined threshold, it informs the other holons in the holarchy of the situation. If there is a "cool" holon that can manage the task in-hand, it starts a negotiation interaction with the hot holon in order to transfer the task. The auto–organization of the entire system is achieved when the holarchies that make up the system try to maintain a temperature equilibrium.

In [17], Brennan and Norrie propose a holonic agent architecture, using agents for the deliberative layer and function blocks for the physical control layer. In [18] a holonic architecture for device control (*HDC*) is illustrated. In Fig. 2.5 we can see the architecture components. The deliberative layer has two purposes: application-domain-specific functionalities, and generic functionalities. The generic functionality is defined by the *planner*, *process model*, *execution control* and *diagnosis* modules. The deliberative layer communicates with the other layers by means of the device *data table* through the *DTACI* (Data Table Access Common Interface). The deliberative and control layers can read and write from and to the *DTACI*. The *control functions* layer is the user-defined application that controls the hardware
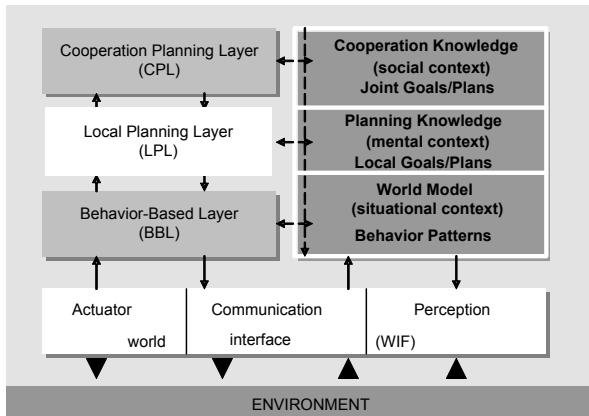
Fig. 2.6 The INTERRAP architecture

and physical process behavior of the *HDC*. On this level function blocks are used. The *physical* layer represents the hardware (sensors and actuators) controlled by the *HDC*. The *simulation* layer is the simulation of the hardware.

The German Research Center for Artificial Intelligence (DFKI) has developed an agent-based architecture to implement holonic system [19]. The architecture is based on the three concurrent layer agent architecture INTERRAP of Müller [20]. Figure 2.6 shows the INTERRAP architecture, in which the composition and configuration of the holonic structures are implemented in the *cooperative planning layer (CPL)*. The *CPL* provides the communication, negotiation and administration functionalities for the holonic structures. They have used this architecture in application domains such as: supply webs, HMS, virtual enterprise logistics and agent-based knowledge source.

In [9] Christensen overviews the holonic architecture approaches most used in the HMS field and proposes as standard an integrated architecture of function blocks and cooperation domains, very similar to the work of Fletcher et al. [12]. For low–level control he proposes the IEC 61499 standard [14, 15], while for high-level control, that is holons' negotiation/coordination in holarchies, he proposes using FIPA agents [21].

2.3.2 Holons Interconnection

The way in which the holons interconnect among themselves defines holon organization patterns that can be useful for modeling and implementing cooperation holarchies using predefined structures.

The group of Mechanics and Manufacturing Engineering of Calgary University has developed many projects related to models for manufacturing systems intelligent

control. Some of these projects are: MetaMorph I, ABCDE (agent based concurrent design environment), DIDE (distributed design environment), FBIICDE (feature-based integrated and intelligent concurrent design system) and MetaMorph II. These projects are based on a factory intelligent control distributed architecture [17, 18].

The main feature of the systems based on MetaMorph [22] is their changing structure. These systems adapt to emerging tasks and changing environments. The MetaMorph architecture uses the domain cooperation concept developed by Deen and Fletcher (see Sect. 2.3.1) but they call it a *dynamic virtual cluster*. In Meta-Morph there are also types of holons or primary holons, as in PROSA (see Sect. 2.3.3). There are product holons, product model holons, and resource holons. A resource holon is dual, on the one hand it has a physical component, which is the product itself from the beginning to the end; and on the other hand, it stores information related to the process status and the product components during the manufacturing process. A product model holon stores the product life-cycle information, configuration data, design specification, process plan, materials list, quality data, etc. The resource holons are used to model manufacturing devices and operations.

The coordination and auto-organization are implemented by means of the dynamic virtual clusters (Fig. 2.7). In a dynamic virtual cluster the holons may participate dynamically in different clusters (holarchies) and may cooperate through cooperation domains. The primary holons fulfill the same task as the coordinator holons of Deen and Fletcher, and are cluster managers coordinating the holons' interactions. The cluster exists while the cooperation task is active, that is, when the task is completed the cluster disappears. The process cycle for a virtual cluster can be defined as: (1) The primary holon joins some or all the contracts (production orders, cooperation orders, etc.) in a new task. After a replanning and analysis processes, the primary holon lists the cooperation requirements as cooperation tasks. (2) A mediation holon is created in order to find a list of potential cooperating holons. (3) The potential cooperating holons are invited to enter the virtual cluster. These holons decide whether or not to participate and send proposals for all the tasks they
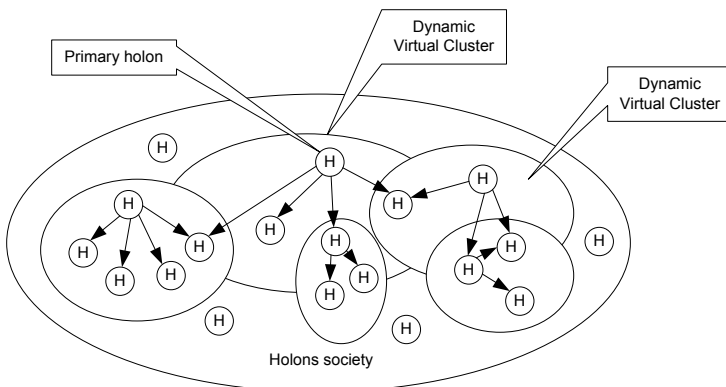

Fig. 2.7 Holons society and holarchy

are interested in. (4) All the proposals are collected and evaluated by the primary holon and the mediator holon. When the optimal tasks assignment is determined, contacts are established directly between the primary holon and the subcontracted holon. The virtual cluster is built among the primary holon and all the subcontracted holons. (5) The associated cluster, the mediator and the cooperation links disappear when the tasks are completed.

2.3.3 Holons Operation

In a manufacturing system there are many types of operation or functionalities that are crucial for the production process. The way in which these operations are identified and modeled is another research and development subject in the HMS field.

PROSA (Product-Resource-Order-Staff Architecture) [23], is the reference holonic architecture for HMS that has been widely adopted. Basically, PROSA is an interholonic architecture, which identifies the types of holons necessary for any manufacturing system, its responsibilities, and the interaction structure in which they cooperate. The architecture is made up of three basic holons, Fig. 2.8: *work-order holon*, *product holon*, and *resource holon*. These holons are specified using object-oriented concepts such as aggregation and specialization. Each of the basic holons is responsible for one of the following manufacturing controlling aspects: internal logistic, manufacturing planning, and resource management. In order to assist the basic holons, with expert knowledge, a *"staff" holon* can be added. The structure of the entire manufacturing system is a dual holarchy divided into one subholarchy of resources assignment (*work-order holons*, *resource holons*, and *staff holons*) and one subholarchy of process control (*product holon* and the *resource holon* controlling process parts). A *resource holon* has a physical part (that is, a production resource of a manufacturing system) and an information processing part that
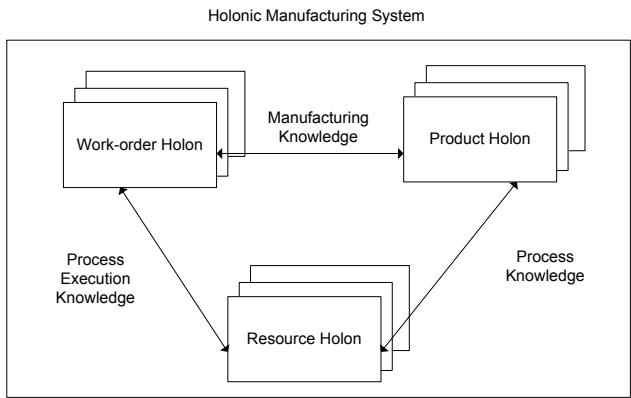

Fig. 2.8 PROSA: reference architecture

controls the resource. This holon offers production capacity and functionality to the other holons. A *product holon* stores the process and product knowledge needed to insure the correct manufacture of the product. It acts as an information server for the other holons in the HMS. A *work-order holon* represents a task in a manufacturing system. It is responsible for doing the work assigned on time and in the right way. It manages the physical products that are being produced, the product status models, and all the logistic processing information related to the task. A coordination and controlling technique for a PROSA holarchy using swarm-inspired social behavior is described in [24].

The ADACOR architecture (adaptive holonic control architecture) [25] proposes a holonic approach for the dynamic adaptation and agility in the face of disturbances in FMSs (flexible manufacturing systems). The architecture is based on a group of autonomous, intelligent and cooperative entities (holons), in order to represent the factory components. These distributed components can be either physical resources (numerical controllers, robots, programmable controllers, etc.) or logical entities (products, orders, etc.). ADACOR groups the holons of a manufacturing system into product holons, task holons, operational holons, and supervisor holons [26]. Each product is represented by one product holon that has all of the product-related knowledge and is responsible for the process planning. The product holon receives the product manufacturing orders. To this end, it stores information about the product structure and the process planning to produce it. Every manufacturing order is represented by a task holon, which is responsible for controlling and supervising the production plan execution. It includes the order decomposition, the resource assignment plan and the execution of this plan. The operational holons represent the physical resources of the factory, such as human workers, robots and machines. They manage the behavior of these resources based on their goals, constraints and capabilities, and try to optimize their agenda. The product, task and operational holons are very similar to the product, order and resource holons of PROSA [23]. The ADACOR supervisor holon is the PROSA staff holon, and is in charge of coordination and global optimization tasks, coordinating various operational and supervisor holons.

The HCBA (holonic-component-based-architecture) [27] is derived from CBD (component-based development) and HMS. HCBA defines two major holons: product and resource. The resource holon is an embedded system component that can execute operations such as production, assembly, transportation, and checking. The product holon may contain a physical part and a controlling part. The physical part can be, for example raw materials, product parts and pallets. The controlling part may represent the path controlling a production line, process control, decision making and product information. The holonic system is built associating these two types of holons, building nested structures of products and resources.

2.3.4 Holonic Control

In this section some works related to algorithm definition and implementation for holonic control are presented. These works can be grouped into four categories of controlling activities: work order programming, scheduling, work-order execution and job-shop control, and device controlling.

- *Work-order programming*: The holonic programming of production operations has been studied in [28, 29, 30, 31, 32, 33]. These approaches usually deal with an interaction scenario in which the product holon is in charge of determining the necessary parts or sub-assemblies and the manufacturing operations. The type of resources associated with every operation and the sequence is determined by means of a set of cooperation interactions with the resource holons.
  The benefits of the holonic approach compared with traditional production programming approaches are due to the distributed nature of the planning process, the interactive cooperation interaction among the production components, and the easy incorporation of new products or resources.
- *Scheduling*: A significant research effort has been devoted on holonic scheduling algorithms. Much of these works focused on flexible manufacturing systems [28, 34, 35], assembly lines [36], jop-shop [37], assembly and machining workcells [38, 39, 40], continuous process lines [41, 42] and factory maintenance [43]. In [32, 44, 45] generic scheduling methods for holonic manufacturing were proposed. The main reason for the large number of activities in this field is the maturity level of the intelligent scheduling techniques [46, 47] and the algorithms for factory distributed control [8, 48, 49], due to the fact that both of them are similar to the holonic approach. They attempt to assign time and resources in a more dynamic way than can be done with offline scheduling methods.
  The major feature of a holonic scheduling approach is that every holon is a problem-solver and a decision-making entity. They use cooperation strategies in order to exchange information and mutually accepted solutions. There is a mechanism to assure that global system constraints are satisfied. And finally, there is a central coordination mechanism.
  The benefits of a holonic scheduling approach compared with traditional approaches are due to the computation and decision-making distribution, and the interactive nature of holons.
- *Work-order execution and job-shop control*: This activity involves the initiation, control, monitoring and termination of tasks and involves actual plans and actual production settings. Work-order execution was studied in [38, 39, 50, 51]. The new elements of the holonic approach in contrast to conventional execution controlling algorithms are [3]: the execution is implemented by means of a negotiation interaction sequence; and the resources (machines) executing the manufacturing operations are responsible for the decision-making regarding the timing and the type of execution.
- *Device controlling*: The device control – which involves actuation, sensing and feedback control of the physical operations that support a machine – has been

largely studied in the HMS field as a conventional control problem [52, 53, 54, 55, 56]. These studies focus on achieving an effective device interface.

2.3.5 Methods for HMS Development

Manufacturing systems are very large and complex. Accordingly, the holonic systems that control or implement them are also large and complex. The development process of these kinds of systems has to be guided by software engineering methods and principles in order to help the engineer in the development process of the HMS itself. To date, almost all of the applications in the HMS field have been built using no design or development method. Moreover, there are probably no two applications developed with the same method. Perhaps this situation has come about because there has been little work done on methods for HMS development.

In [57], *a formal specification approach for HMS control* is presented, but it is still in a developmental stage. There are no defined development phases, and no detailed descriptions to explain how to model issues such as cooperation in the holarchy, holon autonomy and system flexibility. In [58], an agent organization is proposed to model each holon/holarchy that is independent of any holon architecture. However, it is focused only on the holarchy definition and does not define the development phases. The development of HMS is studied in detail in Chap. 4.

There is a definite need to have methodologies for holonic systems [3], that are based on software engineering principles in order to assist the system designer at each stage of development. This methodology should provide clear, unambiguous analysis and design guidelines. In order to fill this gap, we present in this book, ANEMONA, a mutliagent method specifically conceived for HMS development, which tries to fulfill all the HMS modeling requirements.

2.4 Conclusions

In this chapter we have discussed the background of HMS and analyzed the state-of-the-art in HMS. We have attempted to present a global overview of the field, covering the different studies worked on: architecture (Sect. 2.3.1), holons interconnection (Sect. 2.3.2), holons operation (Sect. 2.3.3), holonic control (Sect. 2.3.4), and methods for HMS development (Sect. 2.3.5).

The more active fields are those related to developing holonic control systems. From these developments we can conclude that currently multi-agent system technology is the tool most utilized for developing HMS. Nevertheless, there is very little work on methods for HMS development. The focus of this book is HMS development. To this end, in Chap. 4 we study HMS development requirements indepth and present ANEMONA, a HMS development methodology.