A Taxonomy of Autonomy in Multiagent Organisation

Michael Schillo and Klaus Fischer

German Research Center for Artificial Intelligence,
Stuhlsatzenhausweg 3, 66123 Saarbrücken, Germany
{schillo, kuf}@dfki.de
http://www.dfki.de/~{schillo, kuf}

Abstract. Starting from a general definition of how to model the organisation of multiagent systems with the aid of holonic structures, we discuss design parameters for such structures. These design parameters can be used to model a wide range of different organisational types. The focus of this contribution is to link these design parameters with a taxonomy of different types of autonomy relevant in multiagent organisation. We also discuss the constraining effect of autonomy on the recursive nesting of multiagent organisation. As the domain for applying multiagent systems we choose a general view on multiagent task-assignment.

1 Introduction

The relationship between organisation and autonomy is of increasing importance to researchers from distributed artificial intelligence (DAI). Both concepts are of fundamental importance to the design of multiagent systems. According to Jennings, the "development of robust and scalable software systems requires autonomous agents that can complete their objectives while situated in a dynamic and uncertain environment, that can engage in rich, high-level social interactions, and that can operate within flexible organisational structures" [1]. The advantages of agents that act in organisational structures he sees are that organisations can encapsulate complexity of subsystems (simplifying representation and design) and modularise functionality (providing the basis for rapid development and incremental deployment).

We have previously presented a set of organisational forms for multiagent systems and discussed how they relate to agent autonomy [2]. In this paper, we will take the approach one step further. We will generalise from the concrete forms of organisation and describe the set of underlying design parameters from which these organisational forms (and many others) can be produced. Then we will show how these parameters match to different aspects of autonomy. For this discussion the work of Castelfranchi is important, who showed how autonomy can be founded on dependence theory [3].

Holonic multiagent systems provide the basic terminology and theory for the realisation of multiagent organisation and define the equivalent of modularity

M. Nickles, M. Rovatsos, and G. Weiss (Eds.): AUTONOMY 2003, LNAI 2969, pp. 68–82, 2004.

and recursion of traditional computer science to the agent paradigm. In a holonic multiagent system, an agent that appears as a single entity to the outside world may in fact be composed of many sub-agents and conversely, many sub-agents may decide that it is advantageous to join into the coherent structure of a super-agent and thus act as single entity. These concepts have successfully been applied to multiagent systems especially in the area of distributed scheduling. In order to ground our contribution, we consider particularly this domain, which can range from meeting scheduling, to supply web scheduling, and to service composition in the semantic web.

In the following section we present a definition of holonic organisation that both utilises recursion and allows flexibility in describing forms of holonic organisation. We will use this definition as a framework to describe design parameters for multiagent organisation in Section 3. In Section 4 we identify different types of autonomy and show how they match to these design parameters. Some issues of recursive nesting and agent autonomy are covered in Section 5.

2 Holonic Multiagent Systems – A Framework for the Definition of Multiagent Organisation

The term "holon" was originally coined by Arthur Koestler [4], according to whom a *holon* is a self-similar or *fractal* structure that is stable and coherent and that consists of several holons as sub-structures and is itself a part of a greater whole. Koestler gave biological examples. For instance, a human being consists of organs which in turn consist of cells that can be further decomposed and so on. Also, the human being is part of a family, a team or a society. None of these components can be understood without their sub-components or without the super-component they are part of.

To the outside, multiagent holons are observable by communication with their representatives. These are called the *head* of the holon, the other agents in the holon are part of the holon's *body*. In both cases, representative agents communicate to the outside of the holon in pursuit of the goals of the holon *and* coordinate the agents inside the body of the holon in pursuit of these goals. The binding force that keeps head and body in a holon together can be seen as commitments. This differentiates the approach from classical methods like object-oriented programming: the relationships are not (statically) expressed at code level, but in commitments formed during runtime. For a multiagent system consisting of the set $\mathcal{A}_t$ of agents, the set $\mathcal{H}_t$ of all holons at time t is defined in the following way.

Definition 1 (Holonic Multiagent System). *A multiagent system $\mathcal{MAS}$ containing* holons *is called a* holonic multiagent system*. The set $\mathcal{H}$ of all holons in $\mathcal{MAS}$ is defined recursively:*

– *for each $a \in \mathcal{A}_t$, $h = (\{a\}, \{a\}, \emptyset) \in \mathcal{H}$, i.e. every instantiated agent constitutes an* atomic *holon, and*

– $h = (Head, Subholons, C) \in \mathcal{H}$, *where* $Subholons \in 2^{\mathcal{H}} \backslash \emptyset$ *is the set of holons that participate in* h, $Head \subseteq Subholons$ *is the non-empty set of holons that represent the holon to the environment and are responsible for coordinating the actions inside the holon.* $C \subseteq Commitments$ *defines the relationship inside the holon and is agreed on by all holons* $h' \in Subholons$ *at the time of joining the holon* h.

  Given the holon $h = (Head, \{h_1, ..., h_n\}, C)$ *we call* $h_1, ..., h_n$ *the* subholons *of* h, *and* h *the* superholon *of* $h_1, ..., h_n$. *The set* $Body = \{h_1, ..., h_n\} \backslash Head$ *(the complement of* $Head$ *in* h*) is the set of subholons that are not allowed to represent holon* h. *Holons are allowed to engage as subholons in several different holons at the same time, as long as this does not contradict the sets of commitments of these superholons*[1].

A holon h is observed by its environment like any other agent in $\mathcal{A}_t$. Only at closer inspection may it turn out that h is constructed from (or represents) a set of agents. The set of representatives can consist of several subholons. As any head of a holon has a unique identification, it is possible to communicate with each holon by just sending messages to their addresses. C specifies the organisational structure and is covered in detail in Section 3. As long as subholons intend to keep their commitments and as long as subholons do not make conflicting commitments, cycles in holonic membership are possible (see Example 2 below).

Example 1. Given $h = (\{h_1, h_2\}, \{h_3\}, c_1)$ and an agent k intending to request a task from h. As the head of h consists of two subholons h_1 and h_2, k has two options. It can either address h_1 or h_2. In both cases, the addressed subholons will coordinate the task performance inside of h and deliver the task result.

Example 2. Given $h_1 = (\{a\}, \{b\}, c_1)$ and an agent k addressing a. On the creation of h_1, a and b agreed on commitments c_1 that also explain in which cases a needs to act as the head (e.g. when being addressed in a certain manner or when being requested certain types of tasks). In this case, a can deduce from the way it is addressed, whether it should act as the head of h_1 or just for itself. For the same reason, cycles are not a problem in the definition. Assume we have $h_2 = (\{b\}, \{a\}, c_2)$. If k addresses a, it is clear from the definition of h_2 that a is not addressed as one of its representatives (a is not part of the head of h_2). Whether it should respond for itself or as part of the head of h_1 is decided as without the cycle in the case without h_2).

Definition 2 (Further Holon Terminology). *These notions are not required for the definition of the concept of a holonic multiagent system itself, but make it easier to discuss certain properties.*

[1] At this point we make the assumption that agents *act* according to their commitments. A sanctioning mechanism that punishes incorrect behaviour can build on the commitments made at runtime, as they are communicated explicitly, but is not in the scope of this work.

- *A* task holon *is a holon that is generated to perform only a single task. This notion is opposed to* organisational holons*, which are designed to perform a series of tasks.*
- *Delegation of tasks between two subholons h_1 and h_2 of a holon h as part of working towards the goal of h is called* intra-holon delegation*. If two holons delegate tasks and this collaboration is not part of the goal of an encompassing holon, this is called* inter-holon delegation.
- *Finally, a holon that is not atomic is called a* holarchy*. A holarchy of which all nested subholons have only a single head holon, i.e. a holarchy with a tree-like structure, is called a* hierarchical holarchy*.*

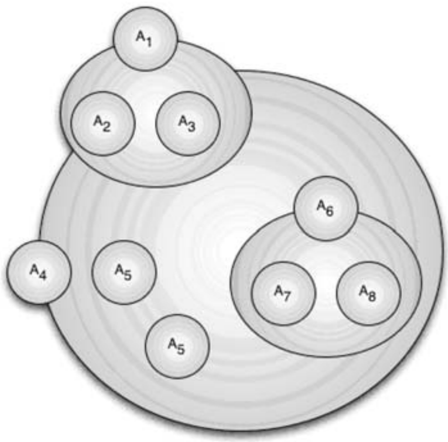

Fig. 1. An example for a holarchy, a complex nesting of holons

Figure 1 gives an example of several possible relationships. The largest entity consists of all agents depicted and has two head holons. The first head holon, named A_4 is atomic, the second one is a composed structure itself and is represented by A_1. The body of the superholon consists of another atomic holon A_5 and another composed holon which is represented by A_6. A_1 has a double function as it represents the holon that represents the outermost holon. A multi-membership is demonstrated by A_7. It is member of two structures (which are both entailed by the biggest holon), in one of them it is body, in the other it is a head member. By definition, all the agents in this diagram could be replaced by further complex holonic structures, resulting in complex relationships as the ones illustrated with A_1 and A_7.

The advantages of the holonic concept are threefold. First, this technology preserves compatibility to multiagent systems by addressing every holon as an agent, whether this agent represents a set of agents or not, is encapsulated. Second, as every agent may or may not represent a larger holon, holonic multiagent systems are a way of introducing recursion to the modelling of multiagent systems, which has proven to be a powerful mechanism in software design to deal with complexity. Third, the concept does not restrict us to a specified type of association between the agents, so it leaves room to introduce organisational concepts at this point.

3 Design Parameters for Holonic Multiagent Systems

A framework that describes the different types of holon defined in Section 2 requires design parameters to span a space for design decisions. In this section we list these parameters, which not only guide the process of creating the concrete holonic structure, but which also restrict the behaviour of each agent and influence the autonomy of each agent inside the holon. They define the set of commitments C that is part of the holon's definition itself.

3.1 Mechanisms for Task Delegation and Social Delegation

Recent work on delegation, has shown that delegation is a complex concept highly relevant in multiagent systems [5,6]. The mechanism of delegation makes it possible to pass on tasks (e.g. creating a plan for a certain goal, extracting information, etc.) to other individuals and furthermore, allows specialisation of these individuals for certain tasks (functional differentiation and role performance). Representing groups or teams is also an essential mechanism in situations which deal with social processes of organisation, coordination and structuring. Following the concept of social delegation of sociologist Pierre Bourdieu [7], we distinguish two types of delegation: task delegation and social delegation. We call the procedure of appointing an agent as representative for a group of agents *social delegation.*

The activity of social delegation (representation) is in many respects different from performing tasks as described previously. For example it involves a possibly long-termed dependency between delegate and represented agent, and the fact that another agent speaks for the represented agent may incur commitments in the future, that are not under control of the represented agent. Social delegation is more concerned with performing a certain role, rather than producing a specified product. In holonic terms, representation is the job of the head, which can also be distributed according to sets of task types to different agents. Just like fat trees (multiple bypasses to critical communication channels) in massive parallel computing, distributing communication to the outside is able to resolve bottlenecks. This makes social delegation a principle action in the context of flexible holons and provides the basic functionality for self-organisation and decentralised control.

Thus, we believe it is justified to differentiate two types of delegation: task delegation, which is the delegation of (autistic, non-social) goals to be achieved and social delegation, which does not create a solution or a product but in representing a set of agents. Both types of delegation are essential for organisations, as they become independent from particular individuals through task and social delegation.

Given the two types of delegation, it remains to explain how the act of delegation is actually performed. We observe four distinct mechanisms for delegation (see also Figure 2):

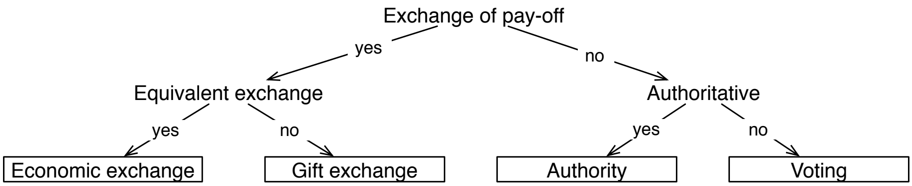

Fig. 2. Overview on four different mechanisms for delegation

- Economic exchange is a standard mode in markets: the delegate is being paid for doing the delegated task or representation. In economic exchange, some good or a task is exchanged for money, while the involved parties assume that the value of both is similar.
- Gift exchange, as an important mechanism in the sociology of Bourdieu [8, pp. 191–202], denotes the mutually deliberate deviation from the economic exchange in a market situation. The motivation for the gift exchange is the expectation of either reciprocation or the refusal of reciprocation. Both are indications to the involved parties about the state of their relationship. This kind of exchange entails risk, trust, and the possibility of conflicts (continually no reciprocation) and the need for an explicit management of relationships in the agent. The aim of this mechanism is to accumulate strength in a relationship that may pay off in the future.
- Authority is a well known mechanism, which represents the method of organisation in distributed problem solving. It implies a non-cyclic set of power relationships between agents, along which delegation is performed. However, in our framework authority relationships are not determined during design time, but at runtime when an agent decides to give up autonomy and allow another agent to exert power. This corresponds to the notion of Scott who defines authority as *legitimate* power [9].
- Another well-known mechanism is voting, whereby a set of equals determine one of its members to be the delegate by some voting mechanism (majority, two thirds, etc.). Description of the mandate (permissions and obligations) and the particular circumstances of the voting mechanism (registering of candidates, quorum) are integral parts of the operational description of this mechanism and must be accessible to all participants.

As suggested by Figure 3, all four mechanisms work for both types of delegation: for example, economic exchange can be used for social delegation as well as for task delegation. This set of mechanisms is by no means complete, however, many mechanisms occurring in human organisations that appear not to be covered here, are combinations of these four mechanisms or variations of their general principles (e.g. different voting schemes). The choice of an appropriate

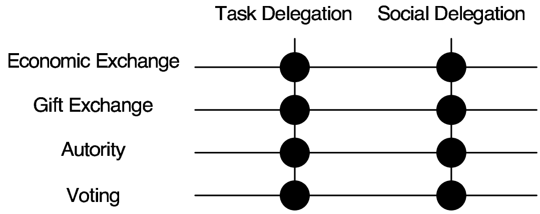

Fig. 3. The delegation matrix showing two modes of delegation and four mechanisms for performing each mode. Theoretically, every combination of mode and mechanism is possible in multiagent organisation

mechanism for the two modes of delegation represents the first two design parameters of a holon.

3.2 Membership Restrictions

A membership restriction can state that there is no limitation or that an agent is only allowed to be a member of one single non-atomic holon. It can also have the value "limitation on product", which means that the holon is free to join another holon, as long as they do not perform tasks of the same type. And, of course, other restrictions are possible.

3.3 The Set of Holon Heads

The number of permitted holon heads permitted is described by another parameter. In very egalitarian holons all subholons can receive incoming tasks and redistribute them, thus all subholons are head holons. More authoritative holons may be organised in a strictly hierarchical manner, with only a single point of access to the outside. An obvious intermediate form is the possibility to define a subset of all subholons as the holon head. The advantage of the egalitarian option is that single-points of failure or communication bottle-necks are avoided. However, the hierarchical option may ease communication with the structure by a single point of access and reduce the communicational effort to coordinate the goals of several holon heads.

3.4 Goals of a Holon

We also allow variations of the goal of a holon. A holon can be created to perform a single task, all task of a single product (i.e. a given type) or it performs all products together that it is able to achieve in collaboration (possibly only making use of a subset of the subholons). This design parameter is central for the interface of the holon to the outside: Only if the goal of a holon is defined, can the prospective subholons determine at creation of the holon if the goals of the holon coincide with any other holon they are member of. In the case of the holon head, a conflict free set of holon goals enables them to determine unambiguously for any incoming request the proper context in which they should process this request. For example, if an agent is head agent for two holons, one of which was designed to process task type t_1 and the other for t_2 and an incoming request matches t_1, then the agent knows it needs to investigate a joint processing of the request in the first holon.

3.5 Profit Distribution

Profit distribution can be done on a per task basis using economic exchange or gift exchange. Other possibilities imply that during the formation phase of the organisation agents agree on how profit is split between head and body agents ("regulation", e.g. 10:90, 20:80 etc.) or that a "fixed income" is being paid from the head to the body agents regardless of the number of tasks performed (in this case, variable costs are paid by the head plus a fixed income chosen by the designer).

3.6 Rules for Inclusion and Exclusion of Subholons

If holons are designed for handling membership flexibly during runtime, they need rules to include or exclude members. We propose two different schemes. The consensus, by which all decision-makers must agree to include or exclude a new member, or the veto scheme where one decision-maker alone veto the majority decision. Furthermore, the set of decision-makers needs to be defined as it need not be the set of all subholons. Reasonable choices are either *all subholons*, *a subset* thereof or *only the head holons*. In case of the exclusion, the single vote equals a veto on the membership: one subholon can then reject the membership status of another subholon.

3.7 Rules for Termination of a Holon

If holons need maximum flexibility, they need at least the possibility to leave superholons, if the membership is no longer beneficial. Already at the start of the holon, these rules can be fixed. Subholons may specify that the termination process is automatically initiated after performing a single task or by veto of a single decision-maker. Other possibilities are that the party intending to terminate the holon pays a fee to the other parties (to compensate for their loss in structure) or the holon is only terminated after a notice period. Any of the last three types requires to specify who is allowed to invoke the process. Reasonable choices are that either all subholons can do this, or only the head holons. In case of a very static system, a holon can be defined to have no option for termination.

3.8 Summary

Table 1 gives an overview over all parameters and their possible values, which shows the complexity of the possible relationships between subholons. The choice from this set of parameters defines the part C of the holon tuple that regulates the commitments among subholons at creation time of the holon. In case subholons are only included later, they are required to agree to the commitments at the time of joining the holon.

Table 1. Overview of the design parameters for multiagent organisation

Parameter	Possible values
Mechanism for task delegation	– Economic exchange – Gift exchange – Authority – Voting
Mechanism for social delegation	– Economic exchange – Gift exchange – Authority – Voting
Membership restrictions	– Exclusive membership – Restriction on product – None
Goal of the holon	– One task – One product – All products
Set of holon heads	– All subholons are head holons – Some subholons are head holons – One subholon is head holon
Profit distribution	– Case by case negotiation – Fixed share – Salary
Rules for inclusion and exclusion	– Consensus – Single vote
Decision maker for inclusion and exclusion	– All or some subholons – All head members
Rules for termination of the holon	– Automatic after task – Veto – After payment or notice period – No termination
Initiator for termination	– Consensus – Consensus among head members – Any member or any head member

4 Agent Autonomy and Multiagent Organisation

As described by Wooldridge et al. [10], autonomy is an integral part of the agent definition. However, autonomy is not a quantifiable notion, but rather consists of qualitatively different types. As Castelfranchi [3] showed, there are several distinct types of autonomy that correspond to different types of *dependency*. Therefore, autonomy of agents is a phenomenon with qualitatively different aspects: an agent can be autonomous (independent) or dependent on others concerning information, the interpretation of information, planning, its motivations and goals, resources, and authority ('being allowed to do X', deontic autonomy) and these dependencies directly relate to losses of autonomy (e.g.

loss of goal autonomy, resource autonomy etc.). Therefore, an analysis of dependencies between agents in holonic organisations will also lead to an analysis of agent autonomy.

Beyond the dependencies in pure agent interaction, new dependencies are created if agents engage in long-lasting holonic structures. The following is a taxonomy for these different types of dependencies between agents, according to the choice of the aforementioned design parameters. These types of dependencies are taken and reformulated from [3], except for *representational*, *exit* and *processing* dependence. We do not share the distinction made by Castelfranchi between *goal-discretion* and *goal-dynamics* dependence, because goal-dynamics dependence is concerned with the timing of goals. As we are concerned with multiagent systems for task-assignment and distributed scheduling, timing is part of the goal description in our context, and (only) therefore the distinction is not relevant here. Also for reasons that lie in our application domain, we combine skill dependence and resource dependence into a single topic, as the resources define the skills in task-assignment.

4.1 Skill and Resource Autonomy

As formulated by Castelfranchi, skill dependence of agent Y on agent X means that the action repertoire of Y is not sufficient for achieving a goal G. Resource dependence of Y on X is the fact that Y depends on the resources of X to achieve its goal G (these resources include time). For our purposes, performing a skill requires the allocation of resources, these two dependencies always come together: if Y needs the skills of X it also requires some of X's resources, requiring resources also requires the skill of using them (it is not envisaged to let an agent surrender its resource to another agent). The formation of holons for the joint performance of a job that requires the resources (and skills) of different agents always includes this kind of dependence. Therefore, all subholons loose to some extent skill and resource autonomy.

4.2 Goal Autonomy

Goal dependence of Y on X is the dependence of Y to choose its own goals. Consider as an example that one would expect artificial agents to have constraints on their goals formulated by their user. As described above, joining a holon includes the definition of a holon goal. If this includes the commitment to perform certain jobs only inside the holon, a participating agent is constrained (and therefore dependent) in formulating new goals: it would be a breach of commitments to perform a job of the same type with a set of agents outside the holon. As these agents can no longer freely choose their goals, they no longer possess goal autonomy.

4.3 Representational Autonomy

Representational dependence of Y on X means that Y depends on X in order to represent it to other agents, either always (Y has no contact to other agents) or only in a special context (X represents Y in specific matters). This type of dependency is probably the most “social” type of dependency, as it does not directly relate to the performance of a task, but only to interaction with other members of the population. It implies loss of representational autonomy, which is of high importance as it deprives an agent of social contacts and may incur loss of opportunities to pursue other interests of the agent.

4.4 Deontic Autonomy

All types of autonomy or dependence state that the agent is allowed to perform some kind of action or not. In the case of representational dependence it is decided at design time of the holon that a body agent is not allowed to represent itself. However, here we are interested in a more complex and flexible mechanism, where we consider cases where the agent is permitted, obliged, or forbidden to do something by some other agent while the holon is active. The expressions, *permitted*, *obliged* and *forbidden* are the three canonical operators of deontic logic as described by von Wright [11]. Therefore, if an agent Y commits itself to wait for X to permit, forbid or oblige Y to work towards a goal G, we call this the creation of deontic dependence. As an observer, we can differentiate deontic dependence from other dependencies by the occurrence of messages between agents that contain one of the deontic operators during runtime (assuming that agents adhere to these kinds of messages).

Deontic dependence means that even if agent Y has appropriate skills and resources, this dependency can stop the agent from pursuing its goals (if it accepts the deontic dependence). Compared to other types of autonomy, deontic dependency translates into a more abstract loss of autonomy. Deontic autonomy has no physical relation to performing a task and is only manifested by the commitments made between agents (i.e. the lack thereof). In our case, this type of autonomy relates to two aspects. First, it relates to the task delegation mechanism “authority”, which means that Y looses deontic autonomy if X can delegate tasks to Y by authority, i.e. to oblige Y to do the task, and permit or forbid Y to perform other tasks in the meantime. Second, there is a connection to the notion of the set of commitments that define a holon, as they can specify the necessity for further permissions or obligations that depend on other holon members.

4.5 Planning Autonomy

If an agent Y relies on X to devise a plan for its actions then Y is plan dependent. If subholons have only one representative, it may make sense to centralise planning and remove planning autonomy from the subholons: In tightly coupled organisational forms, reduction of communication costs can be achieved by this design. Also, if X has authority over Y it needs to be aware of Y’s work-plan in order to decide which jobs to delegate to it. In this case X can (but may not) also devise a plan for Y, which again may be used to save communication. Note that these two cases are not necessarily identical, as one agent may have two superiors that have authority over different resources of a subholon. In case Y depends on the plan devised by X, it has lost planning autonomy and the relevant design parameters are the set of holon heads and the task delegation mechanism.

4.6 Income Autonomy

If Y commits itself to accept fixed payments for providing its services from X, and cannot alter these arrangements (e.g. if there is no exit option for the holon, see Section 4.7), then Y has lost influence over its income. If its income is realised by negotiation, it still depends on others, but it still has a choice and hence its autonomy. The design parameter "profit distribution" deals with this issue.

4.7 Exit Autonomy

All subholons have made certain commitments when they entered a holonic structure. As long as this structure persists, these commitments bind the members and they are not independent in this respect. Being able to exit the structure therefore corresponds to a specific type of autonomy named *exit autonomy.* The mentioned dependencies are described by the design parameters "rules for inclusion and exclusion of subholons" and "rules for termination of the holon". For example, the holon can be designed to terminate after a single task, which leaves no autonomy to the subholons to terminate the structure but instead guarantees a foreseeable end. On the other hand, holons can be designed to be terminated by each member (higher degree of autonomy), by the head members (difference in exit autonomy between head and body members) or by only a single member (small degree of autonomy).

4.8 Processing Autonomy

Processing autonomy expresses the dependency between several agents that decided to merge into a single agent. As any of the formerly different agents surrender all their abilities to process information, this autonomy lost in this process is called *processing autonomy.*

4.9 Summary

Several distinct types of dependencies between the subholons can be identified in a holonic multiagent systems. As Castelfranchi [3] argues, such dependencies are directly linked to distinct types of autonomy. Table 2 gives an overview of these types of autonomy and the corresponding holonic design parameters. Note that "processing autonomy" is the combination of several design parameters.

5 Autonomy and Holonic Nesting

Although the recursive structuring of holonic multiagent systems allows in principle the delegates of organisations to be part of other organisations (as described

Table 2. Overview of the Taxonomy of Autonomy for Holonic Organisation

Type of Autonomy	Relevant Holonic Design Parameter
Skill and resource autonomy	Not applicable in collaborative holons
Goal autonomy	Goal of holon
Representational Autonomy	Set of holon heads
Deontic Autonomy	Mechanism for task delegation
Planning Autonomy	Membership restrictions
Income Autonomy	Profit distribution
Exit Autonomy	Rules for termination of the holon, and Initiator for termination
Processing Autonomy	Goal of holon, membership restrictions, mechanism for task delegation, and rules for termination of the holon

in the previous example constellation), it is precisely the issue of autonomy that imposes restrictions on holonic nesting. Here, the concept of autonomy again demonstrates that it is not a simple scalar parameter but has qualitatively different dimensions. While some dimensions are irrelevant for the nesting of holons, others incur restrictions. Without these restrictions, body agents with a higher degree of autonomy would be introduced to an holonic form that in contradiction requires more restrictions on their autonomy.

A summary overview of the types of autonomy critical for nesting in this sense is given in Table 3. The choice of profit distribution or the set of holon heads for the substructure does not constrain the income autonomy or representational autonomy of the superstructure. All other types of autonomy need to be available on the superstructure as well. If the subholon can freely choose which jobs to perform, then so must be the superholon, otherwise super- and subholon could run into conflicting commitments. It is clear that if the subholon has deontic autonomy then the same must hold for the superholon. If the subholon has planning autonomy, then the superholon must also be master of its own schedule (the same holds for processing autonomy). Exit autonomy must be passed on to the superholon level as well. If not, the structure could face the paradox situation that it has no exit autonomy, but all subholons could use their exit autonomy and then the superholon would in fact no longer exist, without having the autonomy

Table 3. Overview of the types of autonomy that are critical for Holonic Nesting

Type of Autonomy	Critical for Nesting
Skill and resource autonomy	n/a
Goal autonomy	$\surd$
Representational Autonomy	
Deontic Autonomy	$\surd$
Planning Autonomy	$\surd$
Income Autonomy	
Exit Autonomy	$\surd$
Processing Autonomy	$\surd$

to decide to stop existence. Skill and resource autonomy is not applicable to this discussion, as the question of a superholon implies by its very nature that several subholons combine their efforts to pursue a common goal and hence are skill and resource dependent.

6 Conclusions

With this contribution we have advanced the state of the art in two ways. On the one hand we have given a classification of different dimensions of autonomy that extends the previous classification of Castelfranchi [3], for example by the notions of exit or representational autonomy, which we consider essential for the modelling of multiagent organisation. On the other hand, we have shown the connection between this classification and the design parameters for multiagent organisation.

The design parameters describe such crucial properties as membership rules, mechanism of internal delegation and representation, etc. The design parameters describe the design decisions that need to be made to create a holonic structure. Beyond this design framework we also presented a list of options for each of the design parameters. For example, we listed a number of possible mechanisms for terminating a holon. It turns out that the choices for each design parameter are interwoven with the autonomy of the involved agents (e.g. the option to exit an organisation). For an agent it is not only a question of being autonomous or not autonomous, or more or less autonomy (in the sense of a single scalar dimension). As we view the concept, the choices for the different design parameters incur dependencies on several dimensions, which correspond to dimensions of autonomy that define holonic organisation. These dimensions of organisation define the interplay between organisation and autonomy.

Holonic multiagent systems have been identified as a superior mechanism for modelling multiagent organisation. As holonic multiagent systems provide the notion of recursive structuring, it is especially important to pay attention to nested organisations. In general, the nesting of holons is restricted by the autonomy granted to the subholons by the organisational form. Although there are some types of autonomy that are irrelevant in this respect, the general rule is that nested holonic structures may not provide more autonomy than the entailing structure.

Acknowledgements

Prof. Castelfranchi's work on autonomy and dependence was an important source of inspiration. We particularly thank him for his comments on the work in this article. Thanks are also due to our colleagues in the socionics project at the sociology department of the Technical University of Hamburg-Harburg: Dr. Michael Florian, Dr. Frank Hillebrandt and Bettina Fley. This work was funded by DFG under contract Si 372/9-2.

References

1. Jennings, N.: Agent-based computing: Promise and perils. In: Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI-99). (1999) 1429–1436
2. Schillo, M.: Self-organization and adjustable autonomy: Two sides of the same medal? Connection Science **14** (2003) 345–359
3. Castelfranchi, C.: Founding Agent's "Autonomy" on Dependence Theory. In: Proceedings of the 14th European Conference on Artificial Intelligence (ECAI 2000), Berlin, Germany, IOS Press (2000) 353–357
4. Koestler, A.: The Ghost in the Machine. Hutchinson & Co, London (1967)
5. Castelfranchi, C., Falcone, R.: Towards a Theory of Delegation for Agent-based Systems. Robotics and Autonomous Systems **24** (1998) 141–157
6. Norman, T.J., Reed, C.A.: A Model of Delegation for Multiagent Systems. In: Foundations and Applications of Multiagent Systems, Lecture Notes in Artificial Intelligence, vol. 2403. Springer Verlag (2002) 185–204
7. Bourdieu, P.: Sozialer Raum und Klassen. Suhrkamp, Frankfurt am Main (1985)
8. Bourdieu, P.: Pascalian Meditations. Polity Press, Cambridge, UK (2000)
9. Scott, W.R.: Organizations: Rational, Natural and Open Systems. Prentice Hall Inc., Englewood Cliffs, N.J. (1992)
10. Wooldridge, M., Jennings, N.: Intelligent agents: Theory and practice. The Knowledge Engineering Review **10(2)** (1995) 115–152
11. von Wright, G.H.: Handlung, Norm und Intention. de Gruyter (1977)