The Imperfect Hiding : Some Introductory Concepts and Preliminary Issues on Modularity[§]

Giovanna Devetag*

Enrico Zaninotto*

**Department of Management and Computer Science,*

University of Trento

Via Inama, 5 38100 Trento, Italy

devetag@cs.unitn.it, ezani@cs.unitn.it

First draft

Abstract

In this work we present a critical assessment of some problems and open questions on the debated notion of modularity. Modularity is greatly in fashion nowadays, being often proposed as the new approach to complex artefact production that enables to combine fast innovation pace, enhanced product variety and reduced need for co-ordination. In line with recent critical assessments of the managerial literature on modularity, we sustain that modularity is only one among several arrangements to cope with the complexity inherent in most high-technology artefact production, and by no means the best one. We first discuss relations between modularity and the broader (and much older within economics) notion of division of labour. Then we sustain that a modular approach to labour division aimed at eliminating technological interdependencies between components or phases of a complex production process may have, as a by-product, the creation of other types of interdependencies which may subsequently result in inefficiencies of various types. Hence, the choice of a modular design strategy implies the resolution of various tradeoffs. Depending on how such tradeoffs are solved, different organisational arrangements may be created to cope with 'residual' interdependencies. Hence, there is no need to postulate a perfect isomorphism, as some recent literature has proposed, between modularity at the product level and modularity at the organisational level.

♣ Financial support from the Italian Ministry of University and Scientific and Technological Research (project COFIN 99) is gratefully acknowledged.

1 Introduction

The paradigm of modularity was recently used to describe several situations in which it is possible to separate and reassemble parts - in consumption, or in design and problem solving activities - in order to obtain a high degree of variety in a relatively cheap way. It is possible to obtain variety of consumption goods when it is possible to satisfy needs by mixing and matching components; variety of products when it is possible to design them by assembling simple designs; or variety of search paths when this is needed to explore a vast landscape of possible configurations of a set of variables. In all cases modularity permits to separate and reassemble: separation is convenient because it enables to gain efficiency and to decrease complexity; assembly enables to obtain variety from simple components.

Thus described, the concept of modularity appears to be strictly related to the one of division of labour. However, care is necessary in order to distinguish the two concepts and avoid confusion. While division of labour is referred essentially to phases of a production process, modularity is mainly referred to components of a product. Hence, one may have a fully nonmodular product such as, e.g., steel or chocolate, and yet have the production process organised in such a way that there is a high division of labour and a high specialisation of tasks.

This important distinction notwithstanding, there is indeed a close relation between modularity and division of labour. In particular, in our view, modularization is one among several other possible ways *to artificially obtain division of labour* when components are not simply separable. More precisely, modular is a product whose subsets have been *artificially* separated by a process of variety and variability reduction (i.e. standardisation) acting on *special parts* of the product which are called "interfaces". Modularity differs from other devices aimed at attaining task separation. It differs, in particular, from a process of complete standardisation, like the one typical of Fordism, in that modularization requires standardisation on a minimum set of variables (interfaces) while other variables are kept free to change without affecting the stability of the entire system. Moreover modularization is in itself an *artificial and hierarchical* process. This is not to say that modularization can be obtained only inside hierarchical organisations; what we are going to say is simply that modularity needs a design. But, *who* is the designer, who would state priority rules in this process, is a matter of discussion.

In such a view of modularity as a design strategy, we will sustain the view that the choice on how to adopt and implement a modular approach to product design and division of labour is far from being a trivial one, and there are several tradeoffs implied in the decision about the level of modularity that one firm can adopt in its design strategy for a product. In particular, following Prencipe (2000) and Schilling (2000) we argue that modularity is a matter of degree: fully modular systems and fully integrated systems are more ideal types than a concrete reality. Furthermore, we argue that the firm in adopting a modular approach to division of labour must resolve tradeoffs between different types of interdependencies arising from either technical, organisational, or economic factors. A modular design aimed at reducing technological interdependencies can, in fact, give rise to interdependencies of a different nature which then have to be dealt with by appropriate organisational decisions. Moreover, under some circumstances economic and organisational interdependencies may even have a greater impact on the overall functioning of a modular system than sheer technological interdependencies.

We sustain then the need for a theory of imperfect modularity, in which, given the existing constraints on knowledge and rationality, the key problem becomes that of finding a satisficing balance among co-ordination between units devoted to module development, incentive-alignment between the same units, and overall capability to innovate. Furthermore, we sustain, contrary to some of the recent literature on modularity, that there is no a priori need for a perfect isomorphism between modularity at the product level and modularity at the organisational level. On the contrary, there are potentially several ways in which the tradeoffs implied by modularity can be coped with, either by decentralised markets or within hierarchical organisations. As a consequence, there can be a coexistence of different organisational arrangements devoted to the production of similar products.

The paper is organised as follows: section 2 discusses modularity in relation to artefact production and division of labour. Section 3 illustrates the main tradeoffs implied by a modularity approach to artefact production. Section 4 discusses some issues related to the emergence of the architectural design. Finally, section 5 offers some concluding remarks and directions for future research.

2 From division of labour to modularity

2.1 Artefact production and division of labour

To introduce the problem of division of labour and task separation it is useful to refer firstly to the concept of artefact. We make reference to Simon (1996) which defines it as a combination of three aspects: the internal characteristics, the outer environment and goals linking internal characteristics to the outer environment: “Central to [the description of the nature of artefact] are the goals that link the inner to the outer system. The inner system is an organisation of natural phenomena capable of attaining the goals in some range of environments (…). The outer environment determines the conditions for goal attainment. If the inner system is properly designed, it will be adapted to the outer environment, so that its behaviour will be determined in large part by the behaviour of the latter, exactly as in the case of ‘economic man’. To predict how it will behave, we need only ask ‘How would a rationally designed system behave under these circumstances?'. The behaviour takes on the shape of the task environment" (p. 12). It is worthwhile to note two consequences of the adoption of this notion of artefact. The first one is obviously that the actor which has the control of the inner environment must be able to precisely define his goals in order to carry on a transformation on that environment aimed at attaining them. The second aspect is that the working of the artefact depends on the knowledge we have about the possibilities to predict and manipulate the environment. In other terms, an artefact is a subset of variables put under human control in order to obtain a new expected configuration of the same environment: the knowledge we have of natural phenomena permits to manipulate such a bundle of variables (inner system) in order to produce interactions with environmental variables not directly under control (outer system), in order to attain a given goal. If we represent the goal as optimisation, the artefact production corresponds to the solution of an optimisation program P of the general type:

$$P = \min_{x \in X} f(x) \qquad [1]$$

Following March and Simon (1958)[1], division of labour can be interpreted within the general framework of the theory of problem solving. In this light, it is sometimes possible (and convenient) for the problem-solver to reduce the number of variables under his control by dividing them and assigning subsets of them to different decision units. Potential advantages of this solution are clear: their ultimate reason lies in the limited knowledge of the environment. From the point of view of the single decision unit, it is much simpler to deal with a limited number of variables than to evaluate the complex and interdependent results of acting at once over a big number of variables.

However, there are some important caveats to be made.

The first caveat is that, when tasks are separated, decision makers which control a restricted number of variables of the former inner system, could have not only different information but also different goals. For this reason, a multiplicity of independent optimisation procedures might eventually result inferior to the result of the would be procedure of global optimisation.

The second problem relates to the same idea of complexity reduction. When the control over a subset of variables is put in the hands of a specialised decision unit, the outer environment will grow up, and its change will eventually depend not only on variables out of control of the artificial system, but also on variables simultaneously controlled by other decision units, which have become part of the outer environment. Here lies the essence of the problem of interdependence: if we were able to perfectly predict the impact of controlled variables over the environment, taking into account the joint effect of decisions of other units, it would be worthwhile to take the advantages of the specialisation of knowledge and division of labour; otherwise the expected positive effects should have to be balanced against the negative ones. For a long time, discussions about division of labour took for granted that it was possible (and easy) to isolate single parts of a production process (going from the raw material to the consumer). But despite this confidence, the problem is not at all self evident. Sometimes it is natural to separate single parts of a process, due to the technical features of the production process, with clear cut points which isolate phases or clusters of phases. But in other cases the problem is not so simply stated. In an assembly line, for instance, plant designers have to use complex algorithms and to pass sometimes through a trial and error search process to group very small, indivisible components of the process (what are often called "elementary work components") to design a production unit (a set of elementary components that can be worked by a labour unit). The example of division of labour in an assembly line helps to highlight that division of labour often has not an *a priori* content, but it is in itself a *matter of organisation and design*.

We can use the same example to clarify another point. With respect to the notion of division of labour here proposed in terms of delegation of decision responsibility and control over subsets of variables, the traditional, Fordist-type assembly line can be considered a quite special and extreme case, in which a very limited part of the environment is left to the control of a single unit, while all other outer variables are strictly kept under the control of a centralised production program. Indeed, in a traditional assembly line, it is exactly the centralisation of decision power and control that allows tasks to be separated into a complex hierarchy of specialised units.

This complex artificial system permits to bring the division of labour to its limits under the conditions of validity of a centralised design. In other terms, the final labour unit is part of a hierarchy of artificial systems and its outer conditions are kept fixed by a cascade of programs and controls. Hence, in hierarchical systems of this type, most of the complexity is transferred to the design phase, and here lies their potential fragility. The decision power and subsequently the possibilities for local adjustments of the single labour units are very limited: as a consequence, if the design is flawed, mistakes are likely to spread all over the system and require interventions at the top level of the hierarchy to be corrected.

After this digression, we can generalise a bit what said so far. In order to obtain division of labour, the set of decision variables under control of the inner environment has to be separated in subsets, each one governed by a decision maker, so that the problem [1] can be restated as a system P' of decision subproblems.

The way one variable enters in other problems defines the nature of *interdependence*, i.e. how variable values stated by one unit influence a different decision problem and if there is a relation of order among problems.

[1] See also Egidi and Marengo (1994).

Thus defined, the problem of division of labour can be stated in terms of properties of the solution vector of problems *P* and *P'*, both in terms of optimality and computational complexity. As for the first aspect, it should be wondered what value would be obtained for *P* by substituting in *P* the vector obtained when solving the system *P'*. As for the second aspect, it should be asked what is the relative difficulty to select solutions in the two cases. The ultimate goal is in fact to compare problems Π and Π', where

$\Pi = P - C(P)$ and $\Pi' = P' - C(P')$

and $C(\bullet)$ denotes the cost of finding optimal solutions.

2.2 Structural interdependencies

Division of labour consists in leaving the control over decision variables to specialised decision units. It is worthwhile to fully exploit the properties of division of labour when it is possible to govern relationships between decision units.

To govern interdependencies means to give a new structure to the problem, in order to reduce reciprocal influence among decision variables. The most direct way to obtain this result is to separate sets of variables with the higher internal interdependence and the lower connection with other system control variables: isolated subsystems can be left to specialised decision units without affecting the final outcome of the system. This is a process of decomposition in the sense of Simon (1996).[2]

However, in all but the simplest cases, the decomposition pattern is not at all self-evident. Often, in order to reduce interdependencies, problems must be re-shaped to *give them a new structure*, which resembles the old one, but leaves it with a smaller number of interdependencies to be managed. Organisational studies have highlighted several ways to reshape production systems (or problems) in order to deal with interdependencies. Thompson (1957), for example, argues that interdependencies can be managed by isolating and protecting a firm's technical core from environmental turbulence through the creation of organisational units that act as 'interfaces' and 'filters' between the technical core and the outside environment. Standardisation plays an analogous role, both in intra-firm and inter-firm division of labour: standard measures and interfaces allow to divide labour without caring to reciprocally adapt components or phases. Through these methods it is possible to separate complex

[2] It is important to underline that decomposition can be attained simply by rearranging the problem, or with a proper transformation of the same problem. An example of the first case is the transformation of a non linear problem into an additive one: an historical example of this kind of division of mental labour is in Babbage (1835), where account is given of the calculation of mathematical tables in post-revolutionary France. Complex calculations were simply transformed in a proper arrangement of successive additions and subtractions which could be carried on by a chain of specialised workers (or mechanical devices) each one taking as input the calculation made by the previous one.

systems even in very small parcels, each one with a very limited span of control over a restricted number of variables.

Hence, two concluding observations are noteworthy: division of labour in the case of complex artefact production stems from an artificial 'decomposition structure' applied to a given problem. This decomposition structure is far from being obvious and univocal, but it requires a complex design activity. A good decomposition will isolate clusters of tasks, components, etc. having the highest degree of internal interdependencies and being only loosely dependent on other clusters. However, residual interdependencies remain and have to be managed. Furthermore, it is the very process of labour division that creates additional interdependencies, since what has been divided needs to be eventually re-combined. Particular organisational structures, interface standards, standard operating procedures, and others are all means aimed at coping with the management of 'residual' interdependencies. They too, however, need to be designed.

2.3. Modularity as a design structure

At this point we can introduce the notion of modularity as a particular design strategy, among others, aimed at copying with complex artefact production as outlined before. Modularity seems better suitable than other approaches when the adaptation to the demand variety and the speed of search in a given problem space (i.e., the rate of technological innovation) are more important than tight co-ordination. If this is the case, division of labour must occur in such a way that considerable space for local adjustment and experimentation is assured, unlike the case of a traditional assembly line.

Modularity is usually defined as a design approach that allows different modules to be produced independently and with virtually no need for information exchange between the decision units entitled to their production. The independence between modules is in principle assured by the ex-ante specification of interface standards that prescribe how the different modules must be integrated. Restated in more abstract terms, modularity consists in a transformation of the problem space through the reduction of its dimensionality, which is achieved by fixing some variables, (those that form the interfaces) and leaving the remaining variables (those constituting the single modules) free to change.

Let us go a bit deeper in the structure of modularity.

In organisational and social systems, the idea of interdependence and interaction among parts is often translated in terms of *information transmission* or *communication* (Langlois, 2000). In this respect, Parnas (1972) long ago introduced the notion of *information hiding*, referring to the idea that the information concerning a single module must not (and need not) be 'seen' or communicated to other modules. In much a similar way, Baldwin and Clark (1999) distinguish between *visible design rules*, which constitute the architecture of the system and which must be publicly known, and *hidden design parameters,* which concern the design of the single modules and which must not, as such, be visible.

Langlois (2000) calls *modularization* the entire set of visible design rules, which in turn consists of three parts (Baldwin and Clark, 1997): an *architecture,* specifying the set of modules the system is comprised of, together with their functions, a set of *standard interfaces*, specifying in detail how such modules interact and communicate with each other, and a set of *control standards* testing a module's relative performance and conformity to design rules.

What are the stated advantages of modularity with respect to other patterns of labour division?

First, the reliance on a common platform (the architecture) that is assumed to remain stable for a certain amount of time assures an *inter-temporal substitution of components* as long as components from different generations all share the same standard interface (this is what some authors call *modular upgradability*; see, e.g., Langlois and Robertson, 1992). Hence, modular innovation and architectural innovation can be de-coupled and progress independently from one another.

Another key advantage of modularity should be that of eliminating the *non-monotonicity* in the performance contributions of the components to the global performance of the system. This is essential in order for innovative labour to be carried on in a decentralised manner. Let us illustrate this with an example (Marengo et al., 2000): in a highly complex product such as, for instance, an aircraft, an improvement in one of the components (say, the wings) may *decrease* the overall performance of the aircraft (one can imagine as a limit case that the aircraft may crash), due to a lack of tuning in the remaining components to adjust for this change. Hence, a *better* module may cause a *worse* performance in the overall system due to the existence of interdependencies that are not properly understood and managed. With a modular design, interdependencies (at least the most relevant ones) should be dealt with ex-ante and incorporated in the interface specifications (Baldwin e Clark, 2000), thus allowing subsequent incremental innovation on the single modules to be carried on independently.

In other words, the main advantage of a modular approach to labour division consists in the *ex-ante* management of interdependencies versus the *ex-post* and contingent management of interdependencies that characterises integrated systems. In fact, the adherence to common and fixed interface standards allows to substitute to the ex-post adjustment and direct communication needed to reach adaptation, a structural (partly irreversible) design choice. Hence, much of the complexity, like in the case of the Fordist factory, is transferred to the initial phase of design.

However, there is a key distinction to be made between a modular approach and the approach typical of the Fordist factory and, more in general , of all hierarchical methods of artefact production. Indeed, Henry Ford was the first at introducing a neat separation between architectural and component choices. Strict standards for component interchange, joint with tight prescriptions about control are a clear inheritance from the Fordist factory and were methods aimed at incorporating co-ordination in design. Restated in the modern, modularity jargon, Ford was the first at *hiding information*: workers of an assembly line had no need to exchange information in order to make components fit together. Design choices eliminated problems of co-ordination and the need of information exchange and adaptation; an extreme division of labour was obtained by means of a strict separation between architectural choices and operations, where the first ones were aimed at embedding interdependent decisions in the plant design. The distinguishing feature of a Fordist with respect to a modular architecture is that the first one was overextended, leaving then minimal room for adaptation to component producers. As a consequence, any request for new varieties not taken into account by the process designers needed architectural modifications (Gaio e Zaninotto, 1998, Zaninotto 1998): the hierarchical solution of co-ordination problems obtained in Fordist manufacturing models enabled then to greatly increase the attainable level of division of labour, but eliminated independence of actions and room for adaptation to unforeseen modifications of the outer environment. Here lies the key feature of a modular design that crucially distinguishes it from other forms of division of labour like the Fordist factory: the need to find *the minimal set* of variables to be kept fixed, resolving a trade-off between co-ordination effort and adaptation.

In conclusion, modularity is a design strategy that *modifies* a given problem space in order to create multiple, independent search paths and actions. In order to obtain this, the original problem space must be reduced: the variables that enter into the specification of the architectural design are fixed and hence become *parameters* (or constraints) of the new, transformed problem. The new problem space, though less complex, is limited: hence potentially optimal solutions may be left out and never reached within a given architecture. However, the potential variety that gets lost by restricting the space of reachable solutions is recovered through an increase in 'combinatorial variety', i.e, the possibility to obtain arrays of different products by mixing and matching compatible components.

However, as we will clarify in the following sections, interdependencies have different sources and different effects. As a consequence, a design choice aimed at reducing one effect of interdependence (for instance co-ordination problems) could raise interdependencies of a different kind (for instance, goal conflicts between hierarchical tiers). Hence, the decision on whether or not to implement a modular design should take into account the organisational capabilities to cope with all different aspects of interdependence.

3 Managing interdependencies through modularity: the imperfect hiding

3.1 Three trade-offs for modularity management

Let us return to the debated problem of interdependencies. So far we have somehow assumed that interdependencies 'exist out there' in the world or are created through the very process of division of labour. Little, however, is said about their nature or source. A first important source of interdependencies is obviously technology. In the design and manufacturing of high-technology products the functional relations between parts, components, materials are generally complicated and poorly understood. Quite often the process of technological innovation involves a good deal of trial and error learning, costly testing and frequent (plus sometimes very costly) mistakes. It is not interdependencies *per se* that create problems, but our imperfect knowledge of them, combined with our limited computational abilities of taking into account their joint effects. As seen before, modularity is aimed at reducing technological interdependencies between modules, but a first important limit to a complete elimination is given by our limited knowledge and rationality. Such limits, moreover, are clearly more evident the higher the rate of technological innovation on a given product or industrial sector. Hence, a first order of problems arises whenever an imperfect problem decomposition, due to knowledge limits, leaves out relevant interdependencies that are likely to emerge only in the later phases of development of a given product or technology (for example, several unexpected problems in modular software emerge during the final phases of integration and testing, raising as a consequence the costs of these activities).[3]

Interdependencies may also be of a *strategic* nature, whenever they involve decisions the outcomes of which depend on decisions taken by other actors. The distinction between strategic interdependencies and other types of interdependencies, however, is not clear and risks to be misleading if taken literally. In some cases, as for example in the co-ordination problems outlined by Cooper (1999), strategic interdependence arises out of technological interdependence between the levels of inputs of different decision-makers. Even though in these examples the technological relation between the inputs of different labour units is perfectly understood and transparent to all actors involved, there is uncertainty endogenously deriving from not knowing what the other actors' decisions might be, knowing that one's output depends on their as well as on one's own choice. Hence, the relevant point for our purposes seems to be that interdependence creates *uncertainty*, arising either from imperfect knowledge and predictability of future events related to "Nature", or - even in the case in which Nature is perfectly predictable - from lack of knowledge about other agents' actions, which in turn depend on unobservable features such as their preferences, risk attitudes, and so forth. This uncertainty needs to be managed in some way. Besides, strategic interdependencies may be grounded on technological ones, and vice-versa, so that a neat distinction between the two is meaningless.

[3] The same point was recently made, though in more theoretical terms, by Egidi (2000) in the context of organisational problem solving. Egidi argues that organisations, in dealing with complex problems must rely on simplified or 'coarse' representations, in which interdependencies are grouped and categorised into clusters in order to reduce the dimensionality of the problem space to manageable proportions. Such default representations usually have the property of incorporating 'hidden errors' that subsequently may lead the search for solutions astray.

This important caveat notwithstanding, let us discuss two main kinds of strategic interdependence, as they are usually dealt with in game theory. The first are those that involve incentive incompatibility in various forms, occurring whenever the individual strive toward one's goals leads to collectively sub-optimal solutions. The paradigmatic case is well exemplified by the famous Prisoner's Dilemma, and in general by all those economic situations that involve some conflict between individual and collective preferences. In neo-classical microeconomics, the individual maximisation of utility in a competitive market leads to an optimal resource allocation, but only under the assumption that agents are so numerous to not have any chance to influence each other's behaviour; in other words, under the assumption of absence of any kind of interdependence, which sounds as a rather special case indeed (e.g., Levinthal and Warglien, 1999).

The second type of strategic interdependence is given by *co-ordination problems*. Even in the case in which goal conflict problems should be overcome, strategic problems may arise out of a misalignment in individuals' mutual expectations (e.g., Cooper, 1999). In all such cases, the individual incentive to increase one's level of effort in a particular task is a positive function of other individuals' levels of effort. However, in the absence of communication, negative mutual expectations may arise and become self-fulfilling, getting everybody to get locked in a state that everybody dislikes.

How does modularity cope with interdependencies of the type here outlined?
As said before, modularity may eliminate some interdependencies arising from, say, technology, but at the expense of creating "new" and different interdependencies as a by product. Moreover, even all technological interdependencies can never be eliminated completely, leaving out a "residual" whose management must somehow be planned in advance. Hence, there is room for tradeoffs.

Using our frame to understand how modularity impacts on the management of complexity, we can then highlight three main tradeoffs implied by a choice of modular division of labour. The multiple dimensions of interdependencies leave open alternative design choices depending on how these trade-offs are solved; this, in turn, gives room to several organisational features aimed at dealing with residual interdependencies and which can survive at the same time, each one having different properties in terms of dynamic adaptation.

Hence, it is possible to list the following tradeoffs of modularity:

1. between co-ordination problems and conflict of goals;
2. between co-ordination and interdependence in search;
3. between search and goal conflicts.

3.2 Co-ordination and goal conflicts

The trade-off between co-ordination and goal conflicts can be well exemplified by the simple case mentioned by Camerer and Knez (1996) in their game-theoretic approach to organisations. The authors argue that the difference between games like the Prisoner Dilemma (exemplifying a basic problem of free-riding and hence of conflict between individual utility maximisation and collective benefit) and games like the assurance or Stag-hunt game - which on the contrary exemplify a problem of mutual trust - is not so deep as it may seem, and small changes in incentives may easily transform one game into the other. In fact, incentive alignment eliminates the free-riding problem but creates a parallel problem of co-ordination. Camerer and Knez emphasise one direction of the trade-off: the one going from goal conflict problems to co-ordination problems. For instance, it is possible to solve a Prisoner Dilemma like situation by repetition, but this creates co-ordination problems due to the multiplicity of equilibria for the repeated game. The authors argue that organisational problems fall far more often into the second category of games than into the first, due to the existence of long-term relationships between members, which renders co-operation a best response to other players' co-operation.

Modular systems in our view highlight the opposite direction of the same trade-off: from co-ordination problems to goal conflict problems. In fact in order to benefit from division of labour, it is necessary to decompose artefact production into independent production processes by hierarchically separating architectural design from modular production choices. This, as previously stated, may contribute to "solve" the co-ordination problem by *embedding* co-ordination in architectural choices (Sanchez and Mahoney, 1996). This in turn transforms co-ordination costs from variable to fixed and introduces a new kind of division of labour, the vertical one between architecture designer and module developers. This new task rearrangement, however, risks to create goal conflict problems. In fact, how can this new kind of division of labour be managed? Is it possible to design an institutional arrangement that reflects the hierarchy of modularity? The answer would be simple if it was possible to sign contracts by which the specialised firm that develops the system architecture was able to offer to a market of module producers the right to use the same infrastructure. In this case we should observe the emergence of both vertical (the one between architecture owners and module producers) and horizontal division of labour (the one emerging among producers of specialised modules). With a perfect decomposability, problems of goal conflict arising from the need to establish an architecture for co-ordination should be managed by proper contracts: in this case we should expect to find, in the market, firms that develop architectures and interfaces and that control property rights over them, and firms that compete in a market for compatible modules. This is much in the spirit of Langlois (2000) which maintains that the problem of modularity should be simply stated as the best partition over property rights, and the firm should "arise as an island of nonmodularity in the sea [the market] of modularity" (Langlois, 2000) with an exact correspondence between modular partition and the market structure (the vertical and horizontal division of labour between firms).

In short, if it was easy to allocate property rights over the architecture, i.e. if the span of co-ordination was limited and ex ante valuable, the trade off between goal conflict and co-ordination would be solved. In this case we should expect an exact correspondence between modular structure and institutional arrangements, as in Langlois approach: firms, the "islands of nonmodularity", would have the control over architectures embedding interdependencies and/or of single modules while market exchanges would induce the most efficient way to produce compatible components (modules). But this happy world is not always attained.

Two main reasons work against this happy end: the first one lies on the same nature of the trade off between co-ordination and goal conflicts. The second one calls for other facets of interdependencies, and in particular the trade off between static and dynamic aspects of the interdependence management through modularization. We will address the second point in the following sections.

Let us now illustrate the first point. Is it really easy to allocate property rights over the architecture and regulate the access to standard interfaces? In order to do that it would be necessary to completely internalise the externalities embedded in the infrastructure for co-ordination. To this aim, property rights over the architecture should be preserved, thus making the investment in architecture and modules idiosyncratic. As it was pointed out by the principal-agent literature, ex ante investments aimed at creating conditions for independent modular product development have to be "protected" against bad ex post uses, for the reasons deeply analysed by the agency approach to organisations (Myerson, 1982). This not only calls for proper contractual agreements, but also - and more importantly - for the need to put the property of the investment in modules in the same hands of the firm which has the right to propose the contract (Grossmann and Hart, 1986 and Hart and Moore, 1988).

The need to protect the architectural investment then tends to reinforce firm-centered modes of modular production, proprietary architectures and closed access to interface standards. Interface standardisation, and architecture for modularity, only seldom can be managed by the market: more often interface and module production stay inside the firm, which can adopt quasi-market relationships between units, but which still holds the control of property and of residual rights.

Alternatively, as it was empirically verified by Brusoni and Prencipe (2001) in the cases of aero engines and chemical plants, modularity is managed by a complex hierarchical arrangement involving long term relationships between tiers and a special role for firms acting as system integrators. Internal co-ordination is then reached, but at the price of an imperfect solution of goal conflicts between vertically specialised departments.

Moreover, the need to control the access to architecture and interfaces limits the number of users. Co-ordination solutions, as we have seen, have a strong network externalities component, which can be fully exploited when the architecture is open. In this case, module producers with compatible interfaces could extend their market, thus approaching their optimal production size. The enlargement of a market for modules due to the open architecture would decrease costs of production of variety, which can be obtained by mixing and matching, and would fuel the reduction of co-ordination costs needed to manage division of labour. On the other hand, the loss of property rights control reduces (or eliminates) incentives to invest in architecture. The story of IBM and the open architecture for PCs, depicted by Grindley (1995), goes well in this direction. IBM was necessitated to adopt an open architecture and a modular development for its PCs, but this happened at the expense of the profitability of the architectural investment, weakened the position of IBM against the assemblers and gave considerable market power to module producers. The contrast between co-ordination and goal conflict appears then to be, at the same time, a conflict between the benefits arising from the enlargement of the market for modules and the need for firms to keep control over the span of the investment in the architectural design. We will return in brief to this topic, which is of the main importance to understand the future of modular production.

3.3 Search and goal conflicts

The argument highlighted in the previous section is strengthened when considering the dynamic aspects of modularity. As we have seen, a modular architecture is not only a way to economically produce artefacts with a given range of variety, but it also gives room to artefacts evolution, by decoupling architectural from modular innovation. Products within a given architecture should then be able to inherit the best of each generation of module innovation: this is in fact at the root of the real option approach proposed by Baldwin and Clark (2000) to give a value to modularity. A modular design permits also to enhance innovation by *speciation* and *selection* of modules, thus reinforcing the evolutionary change of technological trajectories (Levinthal, 1998).

However, in order to leave modular development to the market, while maintaining a proprietary control over the architecture, the firm acting as principal should be able to anticipate the entire value that the modular design can have for the agent (i.e. that coming from all possible uses of the module in a given structural architecture), making then possible to remunerate the agent on the basis of his marginal contribution to the system innovation. This is not always the case. Substantial empirical evidence on the software industry demonstrates, for instance, that the span of modular development of a given architecture is not clearly definable in advance, making it impossible to state a clear property rights structure. A good example comes from the software industry, notably a highly 'modular' industrial sector. A necessary feature of modular software is that modules be designed so that the software code can be reused by different developers at different times. However, several empirical studies (e.g., Vendelo, 1999) reveal that the reuse potential of modular software is in practice largely unexploited, with consequent extra-costs and time devoted to unnecessary code replication. The reasons for this have to be found mainly in the lack of proper dynamic incentives at developing software code of 'general' applicability beyond its immediate uses. Hence, more generally, the very key advantage of modularity, which is that of eliminating co-ordination-type interdependencies may generate a drive toward sub-optimality whenever the independent and autonomous pursuit of goals by different agents is not made compatible with the aggregate system outcome through some redefinition of agents' dynamic incentives, even in the absence of any inherent complexity of the task at hand and of any technological uncertainty. In this particular example, the lack of proper incentives to software reuse prevents a modular system from fully exploiting the dynamic advantages given by the inter-temporal substitution of components.

In addition, it should be noted that the incorporation of innovation into modules could shift the value from the architecture to modules themselves, whose producers could bargain over the surplus threatening to adapt their innovation to different interfaces. But this possibility reduces the incentives to have a real modular design, and pushes designers to keep the control of modular development, avoiding to separate components development from interfaces. Here again the case of software gives an example: while in principle the different tools for personal productivity are freely mixable under a given operating system, Schilling (2000) highlights a tendency to present software tools in packages composed of applications that are well suited to interact together (i.e., they have a high degree of synergistic specificity) and much less suited to function well with other applications not belonging to the same package, even within the same operating system. This tendency goes actually *against* increased modularization of software, at least as it concerns the final use of the product.

The two faces of the trade-off stand out more clearly when comparing the newly emerged 'open source' paradigm of software production with the integrated, proprietary production adopted at Microsoft. The key feature that allows fast innovation to be carried on in the highly decentralised manner typical of the open source community lies primarily in that the 'open source' mode leaves everything visible and freely accessible by anyone, with the result that errors are promptly detected and the continuous, on-line monitoring and improvement of pieces of code assures a high degree of technical reliability and a high pace of innovation at the same time. Open source hence, is quite the opposite of the information hiding that we have seen as a primary feature of modular systems. The Microsoft paradigm represents, on the contrary, a typical example in which knowledge is hidden and kept in-house by a single leading firm which does not want to lose the economic advantage of its investment, with the subsequent impediment to an effective modular, decentralised innovation.

Empirical evidence clearly shows that the open source paradigm is a radical and novel exception to a fairly widespread rule. In general, the designers of modular systems possess extensive knowledge about the inner characteristics of a product and want to preserve its value. While from a technical point of view the evolution of architectural and of modular knowledge might be separated, the possibility of a conflict of goals between system designers and module developers creates economic interdependencies which are not accounted for in the design based tradition of modularity (e.g., Baldwin and Clark, 2000).

3.4 Co-ordination and search

Technological interdependencies clearly assume crucial relevance in relation to processes of technological innovation within modular systems. In fact, a trade-off typically involves the core feature of modularity, that of assuring fast parallel search over the space of possible solutions (which, in less abstract terms, means fast pace of incremental innovation). A well designed architecture should permit at the same time to decompose activities, thus enabling faster search and learning, and to fix interfaces that simplify the problem of mutual adaptation of parts and components. However, as previously mentioned, to achieve effective, decentralised search it is necessary to start with a "good" decomposition of the problem itself.

Otherwise, decentralised search carried on within an imperfect problem decomposition causes *ambiguity* in the local performance signals and may cause search itself to move away from theoretical optimal solutions and toward inefficient ones (see Frenken et al., 1999; Marengo et al., 2000). Eliminating the interdependencies by adopting a modular design should in theory eliminate such ambiguity and render the overall performance of the system a *monotonic* function of improvements in single components. Nevertheless it remains to be explained how this goal should be achieved in the presence of limited knowledge and bounded rationality.

Another consequence of the limited capability to completely represent the set of interdependencies, is that an evolving system has to face with continuous emerging interdependencies. This is again evident in software production where, also when a modular design is adopted, the phases of integration and testing are of the main importance.

Therefore, modular systems designers are faced with the following trade-off: on the one hand, an extremely fine decomposition would speed up trial and error search but at the risk of lock-in on very inefficient solutions. A coarser decomposition would allow a greater part of search to occur through an extensive and tight co-ordination and communication, but at the expense of a slower adaptation pace. Hence, in the presence of limited rationality, the problem of modularity cannot be simply stated as one of defining the best structural design, i.e., the one permitting an optimal decomposition of the artefact; the problem is rather to know the dynamic properties of different "imperfect" decomposition structures and interface standards. Consider for instance a given problem: what happens when this is separated in different sub-problems (regardless of conformity to "true" interdependencies) in terms of speed of search and goodness of the solution? A fine decomposition, helped by an appropriate structure of interface standards, permits to obtain a quick co-ordinated move toward a coherent set of local optima, but the same structural design hides interdependencies, and thus inhibits the adaptive movement toward the globally optimal solution.

The existence of a trade-off between decomposition patterns, decentralisation of tasks and search efficiency is neatly outlined in a general model of organisational problem-solving by Marengo et al. (2000, 2001).[4] The authors adapt variants of a model of biological evolution developed in Kauffman (1993) to problems of technological innovation and firm competition in complex environments. Innovative activity by a firm is modelled as search over a problem space given by a set of N dimensions which may present varying degrees of interdependence among them, which are only partly known and understood by the firm itself. Competing firms explore different points of the problem space by means of search algorithms characterised by different decomposition patterns. Their main findings reveal that agents endowed with decomposition patterns that reflect the 'true' underlying structure of the interdependencies are the only ones who are always able to discover the 'optimal' solution to the problem, although on a relatively long time scale. If, however, the goal of finding the optimal solution is substituted with a 'good enough' (i.e., *satisficing* in Simonian terms) solution, then it becomes convenient in terms of time to adopt search strategies with extreme decomposition patterns, regardless of the true underlying problem structure.

In addition, when passing from relatively abstract considerations to more concrete organisational settings, other factors have to be taken into account in the design of modularity: for instance, how will an organisation react to an imperfect modularization, which implies the rising of co-ordination costs?

In conclusion, if it is true that modularization (here intended as decomposition) may be the best problem solving strategy in many cases, on the other hand how such perfect decomposition can be obtained in a dynamic setting seems to be a harder problem than the one of finding the optimal characteristics of the single modules. Therefore the decomposition pattern would be the result of a trade-off between the stability of co-ordination patterns (which helps to speed the search), and the organisational ability to manage the emerging interdependencies. Once again, the correspondence between design structures and organisation is not uniquely defined and different organisations and task separation models could survive.

4 Who is the designer?

From the beginning of this article we have argued that the main difference between division of labour and modularity is that, *first)* the former is usually defined over tasks and processes, while the latter usually refers more directly to products, or in general, to the output of labour; *second)* there is a relation between the two in that modularity is an artificial way to produce division of labour, in the presence of complex technologies and limited rationality. This same fact gives reason of the ambiguity of modularity. In the tradition of Smith, Young

[4] With a different approach, Schaefer (1999) reaches similar conclusions, showing that the problem of optimal task partitioning is N-P complete, hence a firm must trade-off with continuous adjustments between modular upgrading and

and Stigler, the extension of the market (hence, an economic motive) is the only constraint to a full division of labour. That means that when the size of the market is sufficiently large as to make it profitable to organise production by means of labour division, task separation, which is thought of as being naturally "embedded" in the product, spontaneously emerges and does not need to be discovered. Knowledge about task separation is, in other words, a public, freely disposable, good.

We have argued, in contrast to this view, that in none but the simplest cases the decomposition pattern is "embedded" in the product. On the contrary, task decomposition is artificially generated and there can be potentially infinite patterns of task decomposition for a given product, some of which may be better than others. This implies two things: first, the design of decomposition patterns and task separation is a complex activity requiring extensive investment in *knowledge*, which a firm may not be willing to make freely disposable to the market. Second, the result of this design strategy, in the presence of the inevitable limits to knowledge and computational capabilities that we have highlighted, will necessarily be an imperfect decomposition, with residual interdependencies that will have to be managed and corrected "along the way".

Hence, the management of knowledge on which modularity is based creates a new pattern of division of labour, the vertical (or hierarchical) one between architectural designers and module developers. This new pattern, which is at the base of artificial task decomposition, is much more in the spirit of Babbage and Marx than in Adam Smith's spirit. From this fact, as we have seen, follow a number of difficult trade-offs, which require several organisational activities to be managed.

While attained artificially, and through a hierarchical separation between architecture establishment and modular development, the "inside a firm" model of development is not the only possibility. Hierarchy here means simply that the premises of decisions of a given level are established outside that decision level (Simon, 1957, 1996). The question of a hierarchy in complexity management opens then the related question about "who" is the architect of a modular system.

Empirical observations present us two different models of a "hierarchy without authority" for modular development (Langlois, 2000; Langlois and Robertson, 1992). The first one is observed when architectures and interface standards privately developed inside firms, are eventually made freely disposable to the market. The second one is given by the many cases of architectures emerging through a bottom-up, decentralised process of spreading through bilateral negotiation or contagion. We can highlight advantages and disadvantages of either pattern. Clearly, the free accessibility of the architecture on the market eliminates the impediments to efficient co-ordination and search patterns previously outlined, and hence it is certainly preferred to a close, proprietary architecture. Hence, one might conclude that architectures that emerge from decentralised processes are the best

---

intergenerational compatibility.

solutions for the tradeoffs we have analysed. On the other hand, some problems of co-ordination still remain, and are at times exacerbated by the very same dynamic of architecture establishment.

The first problem deals with the inherent path-dependent nature of the architectural design, which in this respect assumes all the characteristics of a standard (see Gaio and Zaninotto, 1998, for a review). Many authors point to the fact that once a given architecture gets established, there is a tremendous cost associated with changing it, due to the fact that shifting architecture would imply destroying existing knowledge (see Gavetti and Levinthal, 1998). Architectural changes are typically changes in a tightly coupled system and, as a result, firms have great difficulties adapting to them despite the modest degrees of technical change (Henderson and Clark, 1990).

Hence, given that changing architecture is costly, it is plausible to assume that there is a positive probability that an industry gets locked-in with an inefficient architecture. An empirical question one may ask is whether this is more likely to happen when the architecture emerges spontaneously out of a decentralised network, rather than when it is the product of a conscious, knowledgeable designer such as a leading firm. In principle, it is possible to conjecture that a centralised management of the design process is likely to assure a greater degree of efficiency combined with higher technological degrees of freedom, which are essential in order to allow a reasonable amount of incremental innovation to subsequently take place. This follows directly from the particular nature of the task, which is in itself *not decomposable* and which presumably requires, firstly, a great deal of tight co-ordination and communication between actors, and, secondly, a kind of forward-looking expectations as to what parts of the system will be subject to the most extensive and fast technological change, which is the key element allowing proper technological degrees of freedom to be planned in advance (see Parnas, 1972; Langlois and Robertson, 1992). As we have argued throughout the work, and how empirical evidence seems to suggest, decomposability of products and decentralisation of tasks requires a certain degree of centralisation of *knowledge*. As Prencipe (2000) points out: "decomposed products call for knowledge integrators" (p.174). This in turn would call for the need of having a leading firm retaining in-house some of the key competencies and capabilities required to design an architecture with a high potential of subsequent innovation and to explore different technological trajectories (Dosi, 1982) in search of potentially "better" architectures.

In addition, in the case in which an architecture emerges from a bottom-up process of local market interaction, a further drive towards lock-in to inefficient standards may be given by *network externalities*. Network externalities or other, similar sources of increasing returns (see Arthur, 1989) may in fact arise out of the interaction and quickly drive the market to adopting standard interfaces which may be inferior to others but happened to be the first at being diffused (e.g., Farrel and Saloner, 1986; Katz and Shapiro, 1985; 1986). At least in principle, such a risk should be minimised when an architecture develops internally and only after a certain degree of technological maturity it is diffused to the market (as was the case, for example, with the IBM architecture for PC in the early 80's).

Some support to this view comes again from the history of the automobile industry, as outlined in Langlois (2000):

..."Henry Ford's invention of the moving assembly line and other high-volume production techniques was a remodularization of the manufacture of automobile parts. Ford and his engineers were inventing the process as they went along, and interfaces were necessarily in flux. Because of this, Ford needed authority and scope of action to bring about radical change quickly. By owning most of the stages of production, he was able to experiment with new techniques without paying the dynamic transaction costs of bringing outside players into the process. Once the process of mass production crystallized, however, and change slowed, Ford realized that he could (re)decentralize production, albeit within the confines of the Ford organization" (Langlois, 2000, p. 25).

Such interpretation seems to support the idea that a perfectly modular division of labour works well *after* a given architecture has been established, i.e., after the process of initial design, quick revisions and systemic modifications have somehow crystallised. In other words, the initial design seems to require a sort of systemic vision, while the market may step in and function properly after a decomposition is already well in place and working.

But, on the other hand, it is sometimes possible to leave for free property rights over the architecture and interface standard, when the gains expected from the spreading of standards and interfaces enabling co-ordination are large enough. A good example of this path comes again from the software industry, where free software models of production are based exactly on the idea of sharing information and a set of common architectural rules. Here again a deep effort is needed to empirically study the economic mechanisms that permit to leave to the market the infrastructure for co-ordination.

A further problem arises when passing from the initial development of a system architecture within a given technological path to subsequent, radical changes stemming from the exploration of totally different paths. As said before, this type of innovation is radical and knowledge-destroying (Gavetti and Levinthal, 1998), compared to the cumulative and knowledge-enhancing process of modular innovation and of incremental improvements to a given architecture. The risk of lock-in with an inefficient system is present both in the case of an architecture that gets developed inside a firm and in the case in which it emerges through decentralised market interactions. However, we have argued, when the initial development phase is internalised, there are higher chances that an architecture is designed so that proper degrees of freedom are embedded in it, leaving enough space for subsequent improvements.

But what are the risks of this internalised process for the exploration of different (and potentially superior) technological paths, i.e., for the driving forces towards an industry's 'unlocking'? For the very same reason for which a firm does not want to make its knowledge freely available to the market, the same firm will not want to render it obsolete by investing in research and development aimed at radical innovations. Hence, the need to protect its investment will maintain a firm away from the exploration of totally new technologies. Besides economic reasons, a further reason for this comes from the consideration that an existing knowledge base (i.e., a set of competencies, abilities, etc.) frames and limits the directions in which additional knowledge will be searched and accumulated.

Hence, radical shifts in a system architecture will most likely emerge from other firms which, having no sunk costs to protect yet, may be willing to make risky investments in basic research and development.

5 Conclusions: towards a theory of imperfect modularity

In this article we have tried to present the problem of modularity passing from a engineering to an economic perspective. The economic perspective permits a better understanding of interdependencies, which have not only technical reasons. Moreover, we have highlighted how, to reduce technical interdependencies through modularity, new interdependencies might arise, due to goal conflicts or limited knowledge.

The management of modularity appears then to be, in some sense, always “imperfect”, consisting in the balance of several trade-offs. Different organisations for modularity, giving different weight to one or other aspect of interdependencies, shows different performances when facing environmental instability. This gives room to the co-existence of different organisations for modularity, and the non exact correspondence between the structural features of a problem and the organisation for artefact production. All that remind us to analogous evidences presented more than thirty years ago by J. Woodward (1965), about the relation between structural features of the production process and organisations. A closer, systematic look at organisations for modularity, similar to the one carried on by the British scholar, would shed some light over dynamic properties of different organisations.

6 References

ARTHUR, B.W., 1989, "Competing technologies, increasing returns, and lock-in by historical events", in W.B. Arthur (1994), *Increasing returns and path dependence in the economy*, University of Michigan Press, Ann Arbor, 13-32.

BABBAGE C., 18354, *On the economy of machinery and manufacture*, C. Knight, London.

BALDWIN C.Y., CLARK K.B., 2000, *Design rules. Vol. I: The power of modularity*, The MIT Press, Cambridge Mass.

BRUSONI S., PRENCIPE A., 2001, Unpacking the black box of modularity: technologies, products and organisations", *Industrial and Corporate Change* (forthcoming).

CALLEBAUT, W. (Ed.), 2001, *Modularity: Understanding the Development and Evolution of Complex Natural Systems*, Cambridge: The MIT Press.

CAMERER, C. AND KNEZ, M., 1996, "Coordination, Organizational Boundaries and Fads in Business Practice", *Industrial and Corporate Change*, 5, 89-112.

COOPER, R.W., 1999, *Coordination Games: Complementarities and Macroeconomics*, Cambridge, Cambridge University Press.

DOSI G., 1982, "Technological paradigms and technological trajectories", *Research Policy*, v. 11, 147-162.

EGIDI M., 2001, "Biases in organizational behavior", forthcoming in M. AUGIER and J. MARCH (eds.), *The economics of choice, change and organizations: Essays in memory of Richard M. Cyert*, Edward Elgar,

EGIDI M., MARENGO L. 1994, "Division of labour and social coordination models. A simple simulation model", in GILBERT, CONTE (1995).

FARREL, J., AND SALONER, G., 1985, 'Standardization, compatibility, and innovation", *Rand Journal of Economics*, 16(1), 70-83.

GILBERT N., CONTE R. (eds.) 1995 , *Artificial societies: the computer simulation of social life*, UCL Press, London.

FARREL, J., AND SALONER, G., 1986, "Installed base and compatibility: Innovation, product preannouncement, and predation", *American Economic Review*, 76, 940-955.

FRENKEN, K., MARENGO,L., VALENTE, M., 1999, "Interdependencies, nearly-decomposability and adaptation", University of Trento, Computable and Experimental Economics Lab Working Paper.

GAIO L., ZANINOTTO E., 1998, *Standardizzazione e modelli di produzione post-Fordisti*, CEDAM, Padova.

GAVETTI G., LEVINTHAL D., 2000, "Looking forward and looking backward: cognitive and experiential search", *Administrative Science Quarterly*, v. 45, 113-137

GRINDLEY P., 1995, *Standards strategy and policy. Cases and stories*, Oxford Universty Press, Oxford.

GROSSMAN S.J., HART O.D., 1986, "The costs and benefits of ownership: a theory of vertical and lateral integration", *Journal of Political Economy*, v. 94 (4), 691-719.

HART O.D., MOORE J. 1988, "Incomplete contracts and renegotiation" *Econometrica*, v. 56(4), 755-785.

HENDERSON R., CLARK K., 1990. “Architectural Innovation: the Reconfiguration of Existing Product Technologies and the Failure of Established Firms”, *Administrative Science Quarterly* 35: 9-30.

HOUNSHELL D.A., 1984, *From the American system to mass production: 1800-1932*, John Hopkins University Press, Baltimore and London.

KATZ, M., AND SHAPIRO, C., 1985, “Network externalities, competition, and compatibility”, *American Economic Review*, 75(3), 424-440.

KATZ, M., AND SHAPIRO, C., 1986, “Technology adoption in the presence of network externalities”, *Journal of Political Economy*, 94, 822-841.

KAUFFMAN, 1993. *The Origins of Order*, New York: Oxford University Press.

LANGLOIS R.N., 2000. Modularity in Technology and Organisation, paper presented at the conference on Austrian Economics and the Theory of the Firm, August 16-17, 1999, Copenhagen Business School.

LANGLOIS, R. AND ROBERTSON, P. 1992. , Networks and innovation in a modular system: Lessons from the microcomputer and stereo component industries, *Research Policy*, 21: 297-313.

LEVINTHAL D.A., 1998, “The slow pace of rapid technological change: gradualism and punctuation in technological change”, *Industrial and Corporate Change*, vol. 7, n. 2, 217-247.

LEVINTHAL, D.A., AND WARGLIEN, M. 1999, Landscape Design: Designing for Local Action in Complex Worlds, *Organization Science*, 10, 3, 342-357.

MARCH J.G., SIMON H.A. 1958, *Organizations*, J. Wiley & Sons, New York.

MARENGO, L., DOSI, G., PASQUALI, C., AND VALENTE, M., 2000. The Structure of Problem-solving Knowledge and the Structure of Organizations, Industrial and Corporate Change, 9, 4, 757-788.

MARENGO, L., PASQUALI, C., VALENTE, M., 2001, “Decomposability and Modularity of Economic Interactions”, in Callebaut, W. (Ed.).

MYERSON R.E., 1982, “Optimal coordination mechanism in a generalized principal-agent problem”, *Journal of Mathematical Economics*, v. ? , pp. 67-81.

PARNAS, D., 1972. “On the Criteria for Decomposing Systems into Modules”, *Communications of the ACM* 15(12): 1053-1058 (December).

PRENCIPE, A., 2000, *Divide and Rule: Firm Boundaries in the Aircraft Engine Industry*, Unpublished Ph.D. dissertation, Science and Technology Policy Research, University of Sussex.

SANCHEZ, R. AND MAHONEY, J. 1996. Modularity, flexibility, and knowledge management in product and organizational design, *Strategic Management Journal*, 17:63-76.

SCHAEFER S., 1999, “Product design partitions with complementary components”, *Journal of Economic Behavior and Organization*, v. 38, 311-330.

SCHILLING, M.A., 2000, “Toward a general modular systems theory and its application to interfirm product modularity”, Academic of Management Review, 25, 2, 312-334.

SIMON H.A., 1957, *Administrative behavior*, McMillan, New York.

SIMON H.A., 1996, *The sciences of artificial*, MIT Press, Cambridge Mass.

THOMPSON J.D., 1957, *Organizations in action*, McGraw-Hill.

ULRICH, K.T., 1995, “The Role of Product Architecture in the Manufacturing Firm”, Research Policy, 24.

VENDELO, M. T., 1999, “Organizing for Software Reuse - a research proposal”, Working Paper, Department of

Informatics, Copenhagen Business School.

WOODWARD J., 1965, *Industrial organization: Theory and practice*, Oxford University Press, London.

ZANINOTTO E., 1998, "Standards and standardizations on the eve of a new century", *Standardview. ACM Perspectives on Standardization*, v. 6, n. 2, pp. 90-93.