Modularity in cooperative product development: the case of the MCC ‘smart’ car

Michael Stephan*

Department of Business Administration BWL 01
Institute of Technology and Innovation Management
Philipps University Marburg
Am Plan 2, D-35037 Marburg, Germany
E-mail: michael.stephan@wiwi.uni-marburg.de
*Corresponding author

Eric Pfaffmann

Deutsche Bahn Intermodal
Railion Deutschland AG
Rheinstrasse 2a, D-55116 Mainz, Germany
E-mail: eric.pfaffmann@db-intermodal.com

Ron Sanchez

Department of Innovation and Organizational Economics
Copenhagen Business School
Kilevej 14A – 3rd Floor, DK-2000 Frederiksberg, Denmark
and
Division of Engineering and Technology Management
National University of Singapore, Singapore
E-mail: sanchez@cbs.dk

Abstract: This paper investigates the relationships between the product development processes, modular product architectures and a modular interfirm organisation. We use the case of Micro Compact Car (MCC), the company that developed and produced the ‘smart’ city car and is now a division of Daimler. We examine the ways in which the company applied a revolutionary approach to outsourcing, early supplier involvement and quasi-integration and a ‘modular’ organisation. The paper also suggests some deficiencies in MCC’s initial approach and proposes a framework for assessing architectural capabilities in product development, intended to improve the organisation of collaborative modular development projects.

Keywords: development partnerships; interfirm collaboration; modular architectures; modular organisation; modular product development; outsourcing; quasi-integration.

Reference to this paper should be made as follows: Stephan, M., Pfaffmann, E. and Sanchez, R. (2008) ‘Modularity in cooperative product development: the case of the MCC “smart” car’, *Int. J. Technology Management*, Vol. 42, No. 4, pp.439–458.

Copyright © 2008 Inderscience Enterprises Ltd.

Biographical notes: Dr. Michael Stephan is a Professor for Technology and Innovation Management at Philipps-University Marburg. At Marburg University he is the Head and Director of the research institute Innovation, Strategy and Competition in the Department of Business Administration. Stephan was educated at the Universities of Basel (Switzerland) and Hohenheim (Stuttgart/Germany) where he graduated in Economics and received his PhD in Management. His research interests include the strategic management of technology and innovation in MNCs, open source issues in intellectual property management and innovation management in services.

Eric Pfaffmann is the Senior Project Manager and Head of Maritime Networks at Deutsche Bahn Intermodal (Railion Deutschland AG) and Senior Research Associate in the Department of Business Administration at Philipps-University Marburg. He was educated at Johann Wolfgang Goethe-University, Frankfurt am Main, where he graduated in Business Administration. He received his PhD in Management at Hohenheim University (Stuttgart, Germany). His research interests cover fields such as strategic management of technology and innovation, theory of the firm, and modularisation of technical systems and organisations.

Professor Ron Sanchez has degrees in Humanities and Science, Architecture, Engineering and Technology Strategy (PhD) from the Massachusetts Institute of Technology, USA, as well as an MBA (with Honors). He has taught at the University of Illinois (Urbana-Champaign), the University of Western Australia, ESSEC (France), IMD (Switzerland) and Lund University (Sweden), as well as in numerous graduate and executive programmes around the world. His research interests include modularity in product, process and knowledge architectures, knowledge management, competence-based management and options theory. He has published numerous papers in leading management, marketing, economics and design journals and several management books. His book, *Modularity: Strategy, Organization, and Knowledge Management* (Oxford University Press), is forthcoming in late 2008.

1 Introduction

In April 1994, Helmut Werner, the then-CEO of Mercedes-Benz, and Nicolas Hayek, the CEO of Société Suisse de Microélectronique et d'Horlogerie SA (SMH), the company that made the world-famous 'Swatch' line of products, agreed on a rather unconventional joint venture in the automobile industry. A new company would be established to develop, produce and sell a highly innovative new city car. Micro Compact Car (MCC), the name selected for the company, would develop a revolutionary small city car, to be branded 'smart'. The name was contrived from taking 's' for Swatch and 'm' for Mercedes and using 'art' to highlight the inventiveness of the total concept (Pfaffmann, 1998b). Operating with limited resources and stiff financial and human capital constraints, MCC was forced from the very beginning to concentrate on the essentials. 'Reduce to the max' was both a marketing slogan and the motto the company used to describe MCC's product development and outsourcing strategy. In part, as a result of its resource constraints, MCC sought to create a new 'modular' organisational form, in which MCC would coordinate a group of key suppliers who would provide more than 85% of the value added of smart.

The smart car constituted a significant departure from the usual product concepts offered by Mercedes-Benz which, of course, has traditionally been focused on luxury cars (Bensaou and Pfaffmann, 1998). The car was intended to create a new market segment in the city car niche. It was innovative in both its technology and the way it was designed and produced. Another key feature of the two-seater smart design was its customised design concept. A customer could customise his or her own vehicle by choosing from a 'menu' of body panel and body colours, which could also be changed very quickly at a low cost anytime after the purchase of the car.[1]

In this paper, MCC's approach to the early supplier integration and organisation of product development will be examined. In particular, the interrelationship of the product design concept for smart and the cooperative interfirm organisation design for product development will be illuminated. We will also outline what we regard as the deficiencies in MCC's conceptualisation and implementation of a modular approach. Drawing on MCC's experiences, we will also propose a framework for organising cooperative product development around modular designs. The paper will evaluate what early supplier integration may imply for the overall organisation of product development and how organisation design relates to organisational learning and capability development in product development. Looking beyond the case of MCC, we will also explore the broader issue of how outsourcing the component design in the concept development stage to upstream suppliers should be managed to assure the smooth integration of physical components in product designs and the suppliers in organisation designs.

We will begin our discussion with definitions of the product development processes and the concept of modular product architectures (Section 2). In Section 3, the paper draws on the ideas of modular product architectures to describe a modular organisation. Section 4 draws on the experiences of MCC's product development process to comment on the important aspects of knowledge management during modular product development. The section also introduces a concept of how to assess the degrees of capability in modular product development and suggests a capability-based typology of the components that may be transferred between 'system integrator' companies, such as MCC, and their suppliers.

2 The role of product architectures in product development

2.1 The product development process of micro compact car

Product development is the sequence of activities that develop and produce an industrial product. The process starts with the identification of customer needs and the generation of product ideas to serve those needs and it ends with product solutions that can be manufactured (Ulrich and Eppinger, 1995). Product development processes can be decomposed into different *stages*, with distinct activities in each stage[2] to provide structure and specify the content in the complex task of developing a new product. Although the completion of a particular stage of development is marked by certain milestones, successive stages very often overlap (Clark and Fujimoto, 1991; Whitney, 1993), especially if information about downstream development is required to perform tasks in upstream stages.

A development process also serves to structure *communication* among the people involved in concept definition, design and development. In the case of smart, the definition and implementation of the product development processes were a particularly delicate and pivotal concern, since the new car concept, the new development process and the substantial role of the external suppliers in the development process called for new approaches to organise development and production. Furthermore, precise development process definitions were of prime importance because they structured day-to-day interfirm task coordination and thereby replaced much hierarchical coordination between MCC as the system integrator and its supplier companies.

Figure 1 illustrates the structure and chronology of the development process of smart. MCC defined the overall development, realisation and production phases, as well as the relationships with the supplier companies. The overall product development process was broken down into six phases:

1 strategic product planning

2 procurement marketing

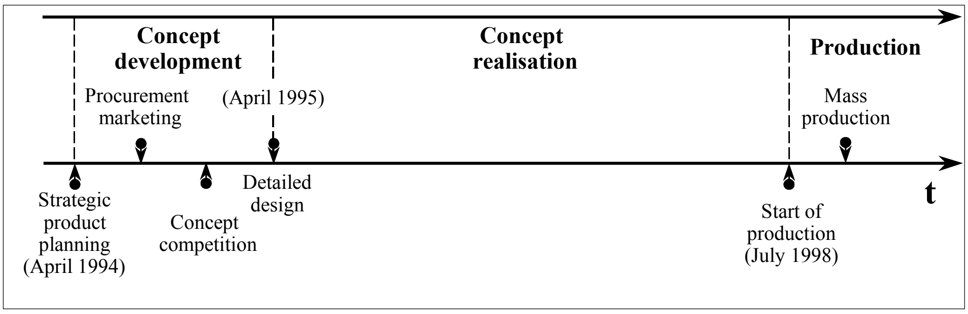
3 concept competition

4 detailed design

5 the start of production

6 mass production.

Figure 1 The structure of the development process



Each phase of the development process had clearly defined targets for both product development and supplier integration into the development and production processes. Furthermore, each phase defined a set of subprocesses that prescribed 'what to do when' and, thus, how to reach the planned targets.

The concept development phase began with strategic product planning to develop an initial concept of smart that would meet the identified customer needs. Customer-oriented target costing linked major product functions to the price the customers would be willing to pay for the functions, *i.e.*, it transformed the economic notion of 'utility' and the marketing concept of customer benefits into the parameter of 'money'. The company also screened the potential supplier companies to determine a preselection 'short list' of suppliers. The supplier companies that successfully survived the initial screening were invited to participate in the concept competition phase.

The concept competition phase consisted of procurement marketing, in which MCC explored the suppliers' creativity and assessed the specialists' domains of expertise. The quality of a supplier's concept study, company size and performance, manufacturing locations, quality certifications and customer references were used as the supplier selection criteria. In effect, the concept competition phase determined who would be MCC's development and production partners and stipulated the contractual arrangements and target costs agreed by MCC and the selected suppliers. The conclusion of the concept competition phase closed the window for fundamental conceptual changes in smart and strongly influenced the subsequent path of the project.

The concept realisation phase began with a detailed design in April 1995 – only one year after the start of concept development – and significantly transformed the project with respect to size and costs. From that point on, the suppliers had to allocate significant human and financial resources to the project and became actively involved in translating the concepts into detailed designs that could finally be manufactured on a mass production basis. We address below some of the challenges MCC and its suppliers faced in using modularity concepts in this key process.

2.2 The concept of the modular product architecture of smart

In product development, defining and implementing a new product concept requires defining all the specific functions of a new product, relating the defined functions to specific kinds of physical components and defining the interfaces between the components that will assure the smooth functioning of the finished product (Sanchez, 2000). These activities, when carried out in a coordinated manner, define the *architecture* of a product.[3]

Broadly speaking, a product architecture is a "...scheme by which the function of the product is allocated to physical components" (Ulrich, 1995, p.420). More precisely, an architecture is a decomposition of the overall functionalities of a product into a set of *functional components* and the specification of the *component interfaces* that define how the components will interact in the product design as a system of components (Sanchez and Mahoney, 1996; Sanchez, 1999).

An architecture becomes *modular* when at least some component interfaces are specified to allow the substitution of the component variations into a product design without having to make compensating changes in the designs of the other components (Sanchez, 1999; 2000). A modular product architecture is characterised by a relatively high *independence* of the functions and physical components in the product design (Göpfert, 1998), which implies that each function is implemented by *one* component – what Sanchez (1999; 2000) calls a 'one-to-one mapping' of functions to components (or to systems, in more complex products).

If the degree of modularity in a design is low, a given functionality may be executed by a number of components and this high interdependence of the design components leads to a high interdependence in the development and production activities (Sanchez and Mahoney, 1996).

Within modular product architectures in the automobile industry today, it is common to physically group components into 'modules' and systems. A *module*, in automobile industry terminology, refers to a subassembly of components that will occupy a given spatial area of the car, *e.g.*, the cockpit, the doors, the seats or the drivetrain. Modules are defined in this way to facilitate the assembly process and minimise the assembly and logistics costs. Within these assembly modules, there may be components from one or more *systems*, *e.g.*, the air management system, the braking system or the power system (Pfaffmann and Stephan, 1999). Systems, the *functions* they have to fulfil and the components that will execute the various functions are defined in the early phases of the product development process. When a system executing a given function is distributed across multiple assembly modules, the development process must take into account both the design requirements for the system itself and the requirements of a module-based assembly process.

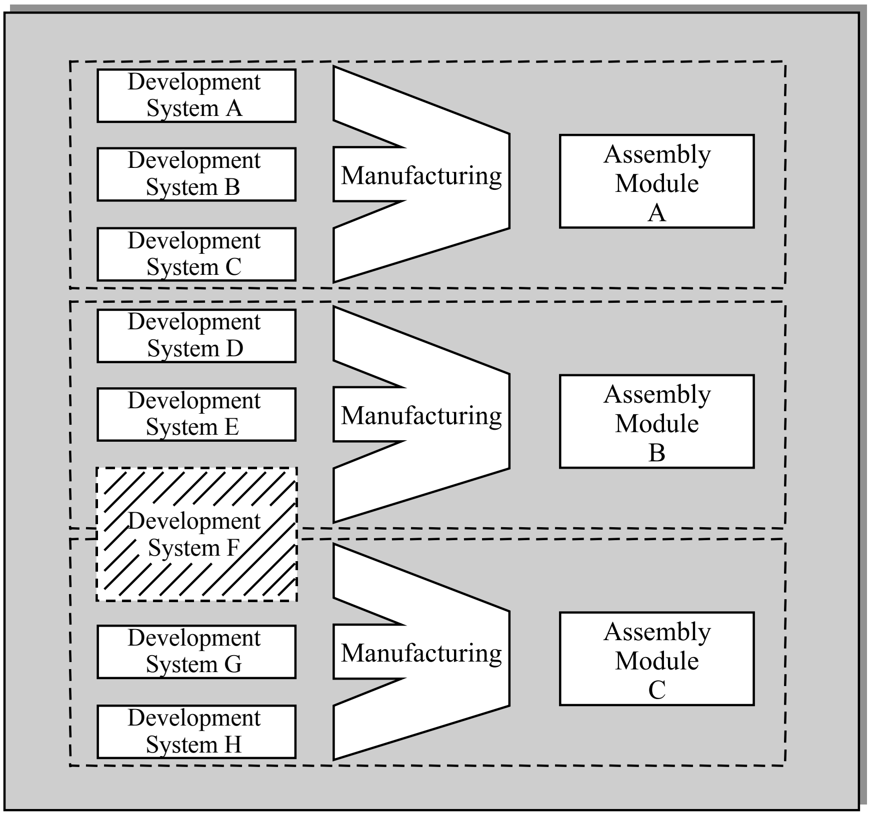
MCC's strategy of outsourcing more than 85% of the development and production to selected suppliers meant that MCC had to carefully define the architecture of smart to balance two interrelated concerns. First, the decomposition of smart's design into functional systems and assembly modules had to be done so that development and production could be 'packaged' in a way that would attract the qualified supplier companies. Second, the development and production packages had to be defined to minimise the number of direct module suppliers. The desire to limit the number of first-tier module suppliers followed from the fact that MCC had extremely limited resources in general and, in particular, did not have the needed management resources to coordinate large numbers of independent parts supplier companies during development and production.

At the same time, MCC wanted to attract suppliers with leading expertise in the specific kinds of modules and systems that smart would require. MCC wanted suppliers that could both develop and produce their respective 'packages'. Thus, in the product architecture of smart, the assembly modules and functional systems would have to be defined so that the functional systems and their interfaces would be contained *within the boundaries* of each related physical module. In effect, the interfaces of an assembly module would have to coincide with the interfaces of the system(s) it contained. Thus, the sourcing packages MCC defined for smart were for *system-modules* (Pfaffmann and Stephan, 1999).

This is illustrated in Figure 2, where assembly module A is a system-module because systems A, B and C are fully contained within assembly module A. Assembly modules B and C are not system-modules, however, because system F is not wholly contained within either assembly module, but is shared between them. MCC was able to define and outsource to its supplier companies a number of self-contained system-modules, such as the cockpit (instruments and dashboard), car frame, seats, complete wheel and tire subassemblies and the Customised Body Panel System (CBS). In other cases, MCC could not define system-modules and had to outsource 'incomplete' assembly modules that lacked some of the components needed in a system that was not fully contained in an assembly module. In the latter case, MCC believed that systems like the brake and suspension systems could be outsourced to suppliers in the development phase and be easily integrated into the assembly modules by the other suppliers in the production process.

MCC believed that a high degree of modularity could be achieved in defining system-modules and that would be the key to the delivery of additional value and utility in two respects. First, the outsourcing of system-modules to the suppliers would allow the concurrent, quasi-autonomous development of smart (Pfaffmann, 2000), which would shorten the development lead times (Sanchez and Mahoney, 1996). In principle, if the interfaces of the modules are clearly and fully defined, the suppliers can focus on developing their system-modules without having to manage integration with the other modules. Hence, achieving modularity in the system-module architecture of smart was a requirement for decoupled *modular development processes* and negligible cross-supplier task interdependencies. Moreover, suppliers could *develop and produce* some system-modules and thereby maintain sole responsibility for their part in smart during the lifetime of the car.

Figure 2 An illustration of a modular product architecture: systems, modules and system-modules

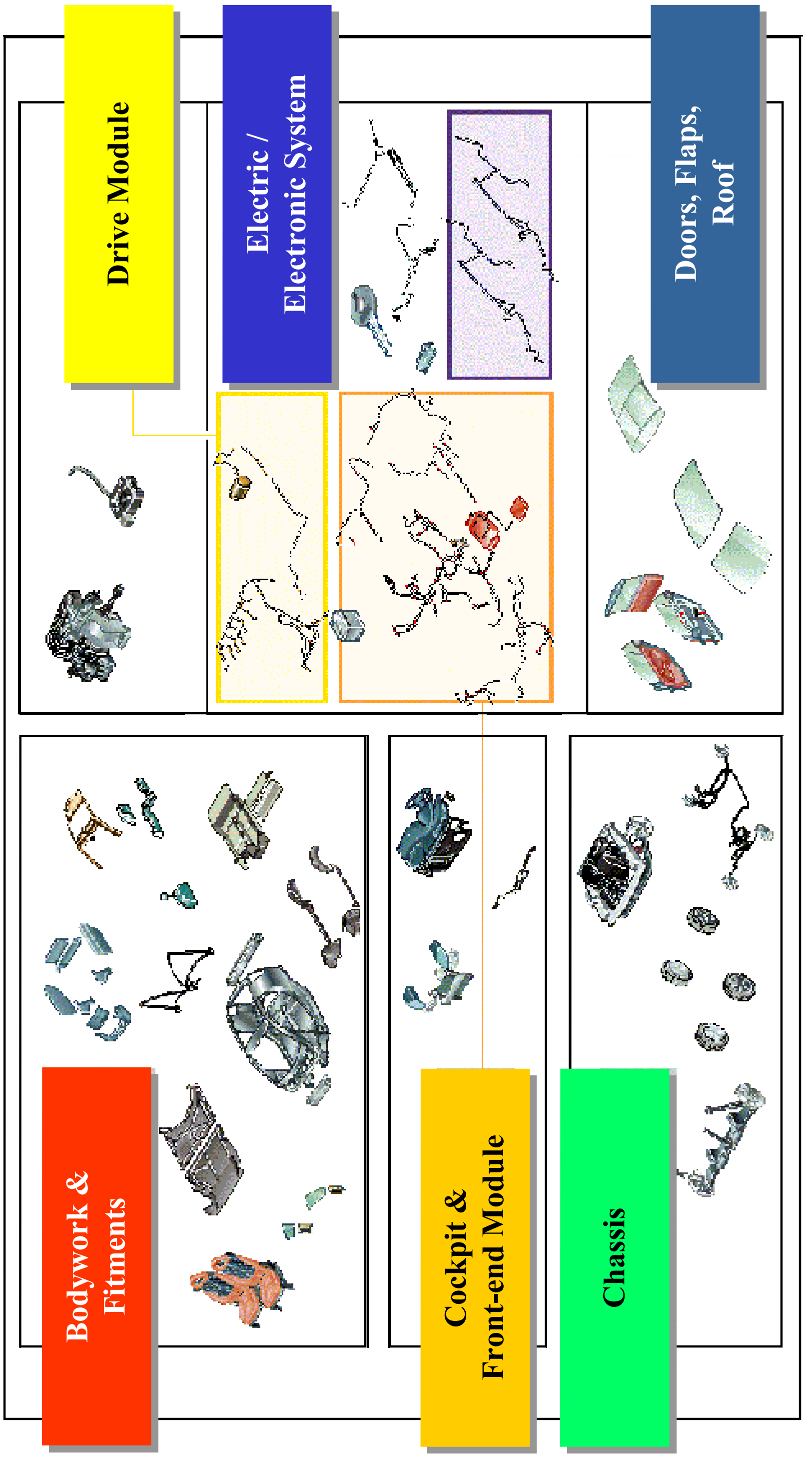


3 Modularity of the product and organisational architectures: the modular organisation of Micro Compact Car

3.1 The relationship between product architectures and organisational architectures

An important consequence of fully defining the interfaces between the system-modules and thereby decoupling modular development processes for individual system-modules is the opportunity to achieve a modular organisational structure that draws its organisational boundaries according to the technical module boundaries and, thus, reflects the design of the product architecture (Sanchez and Mahoney, 1996; 2001; Pfaffmann, 2000).

Figure 3 An illustration of the organisational modules of smart (see online version for colours)



Source: Pfaffmann (1998b)

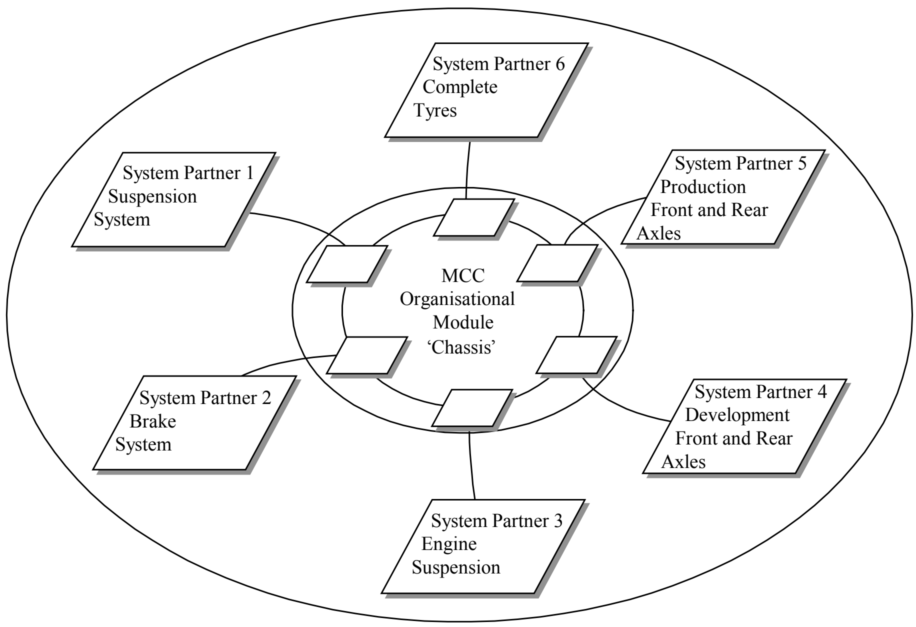
At the beginning of the smart project, MCC defined five organisational modules which aggregated the technical systems and modules of the vehicle (see Figure 3). These organisational modules were:

1 the car body and fittings
2 the cockpit and front module
3 the chassis
4 the drive module
5 the doors, roof and flaps.

(The electric/electronic system shown in Figure 3 is discussed below.) Consequently, each organisational module was to coordinate the development processes of the relevant system partners' technical system-modules within each of these five areas.

At the beginning of the concept realisation phase, MCC asked the selected supplier companies to set up a project organisation and to nominate team members. In effect, MCC structured this project organisation and its own in-house integration activities on the basis of system-modules and the company achieved an *interfirm project organisation* that was isomorphic with the technical system-modules of smart. If the architecture of smart was defined to achieve modularity among the system-modules, modularity could be transferred to the organisational domain and thereby minimise supplier interdependencies during development and production. Figure 4 illustrates this isomorphism in the technical and organisational structures for the chassis of smart and the resulting interactions between MCC and its supplier companies.

Figure 4 An illustration of the isomorphic interfirm system-module organisation for chassis development and production



3.2 The deficiencies in smart's technical and organisational modularity

During the concept realisation phase, MCC began to recognise some deficiencies in the desired degree of organisational modularity and understand that these organisational deficiencies directly reflected the limitations in the degree of modularity achieved in its system-module product architecture. In fact, each system in smart did not fit neatly into its allotted module and several systems 'trespassed' over boundaries with other modules.

Achieving perfect modularity in a system-module architecture is an extremely ambitious objective for an automobile design, since there are several systems (such as air management, acoustics and electrical connections) that normally traverse several areas of a vehicle and, thus, cannot be effectively assigned to a single module unless radically new design concepts are developed. For example, to provide heat for the passenger compartment of a car, the conventional concept is that the car engine is used to produce a secondary function, heat. However, an engine will transmit varying temperatures into the passenger compartment, depending on its own temperature. In addition, air flow into the car interior is determined by factors other than the heating system, such as the speed at which the car is running or the local weather conditions. The heating system itself, therefore, cuts across several areas of a vehicle and cannot be perfectly modularised as a system-module within the conventional automobile design paradigm (Dosi, 1982; Durand, 1992; Henderson and Clark, 1990). Failure to achieve system-module modularity for such systems would likely require redefinitions of the product architecture, leading to *boundary and interface changes* in other system-modules during the course of the development process.

The integration of such systems, which we refer to as *cross-sectional systems*, is difficult because complex interactions with other system-modules tend to be difficult to manage (Sanchez, 2000). At MCC, the development issues arising from the cross-sectional nature of the heating function soon revealed that nobody had been made responsible for its proper integration. Both the suppliers and MCC had been occupied with and pursued the optimisation of their specific system-modules. Only during development did MCC realise that several cross-sectional functional systems of smart were not receiving the same degree of attention as the fully modular system-modules. Furthermore, the modular organisation, premised on the full modularisation of the system-module architecture, did not provide structures for managing cross-sectional systems. In order to address this design issue, MCC added a sixth organisational module, called 'electrical connections/electronic systems', which focused specifically on cross-sectional systems within smart's imperfectly modular system-module architecture. Moreover, the company also established an overlay structure, the organisational cross-sectional module called 'total vehicle optimisation/vehicle testing', to coordinate information flows about cross-sectional systems between organisational system-modules. 'Total vehicle managers' became responsible for the integration, optimisation and testing of the entire vehicle. They had the same hierarchical rank as the functional system managers within the system-modules and their role was to settle intra-MCC conflicts among system-module development groups.

The problem of insufficient modularity in defining the system-module architecture of smart was not confined to cross-sectional systems. At the beginning of the concept realisation phase, the definition of the product architecture of smart did not contain sufficiently detailed and precise specifications of the functions, systems, components and interfaces. Early in the project, MCC felt that it did not have sufficient experience with smart's innovative vehicle architecture to create the initial specifications that would remain stable during detailed concept realisation. In addition, the use of basic descriptions (rather than detailed technical specifications) was thought of as necessary to maintain a high degree of freedom in concept development, but the lack of technically complete definitions and interfaces jeopardised the project once the approximate description of the design was 'frozen' and the modular network started to work.

Subsequently, due to the technical difficulties encountered in the development of interacting system-modules, the start of production, originally scheduled for summer 1998, had to be postponed for a couple of months and an additional DM 500 million had to be invested by Daimler-Benz to resolve a number of the resulting design issues. Major complications occurred in the chassis of smart. When smart failed the now-infamous 'moose' test – tipping over while quickly turning to avoid a moose – the chassis of smart had to be widened to improve lateral stability. This alteration was a significant technical change in smart's architecture, entailing numerous subsequent changes in systems and modules other than the chassis. This suggests that MCC did not anticipate all the performance requirements and resulting technical specifications of smart that were needed to fully define the vehicle's architecture at the beginning of the concept realisation phase. The ensuing redesigns of the system-modules created difficulties within the modular development organisation and led to costly renegotiations with the supplier companies.

A root cause of these development problems was that the *realised* product architecture of smart was not as modular as it was *intended* to be during concept development. Although MCC pursued an ambitious product concept with an extremely high degree of modularity in smart's architecture and its outsourcing approach, the company failed to assess the degree of modularity it was creating in its early description of smart's product architecture. In effect, MCC did not systematically evaluate its own *capability* in defining a modular architecture for smart. Although MCC knew that it would not have the in-house capability to define and detail all the necessary component designs, the company did not realise that it lacked the sufficient *architectural* capability (Sanchez, 2000; 2001) to precisely evaluate and specify during early concept development the degree of modularity required in concept realisation. This was a major cause of unforeseen redesigns during the subsequent concept realisation.

4 Overcoming the deficiencies: the capability assessment and transformation of physical architectures into bundles of capabilities

In the remaining parts of this paper, a conceptual foundation for assessing capability in product development will be outlined and a distinction will be made between *architectural capability* and *component capability* (Sanchez, 2000; 2001). The importance of architectural capability in product development – particularly for innovation and supplier integration – will be delineated. Furthermore, architectural and component capabilities will be linked to key activities in product development and a concept for assessing the degree of these capabilities in product development will be introduced. Finally, a capability-based typology of components[4] will be sketched.

4.1 The knowledge-based foundations of architectural and component capabilities

The distinction between architectural capabilities and component capabilities reflects two processes in product development – the development of *components* on one hand and the specification of an *architecture* that defines how components interact in a design on the other hand (Sanchez and Mahoney, 1996; Sanchez, 2000; 2001).

The development and production of a component requires in-depth knowledge that may span across several domains of technological knowledge that are related to the functioning and/or production of a given component. The ability to design, develop and produce a reliable component is termed here as 'component capability' (Henderson and Cockburn, 1994; Pfaffmann, 2000). *Component capability* is primarily based on substantive knowledge, much of which is *practical knowledge* about how to produce a particular component (Pfaffmann and Scheuble, 1998). To a large extent, such substantive knowledge is rooted in the skills and experiences of individuals that they have accumulated in the course of their professional lives. In most organisations, component-specific substantive knowledge is not perfectly articulated or codified and, therefore, remains fundamentally *implicit* and difficult to transfer (Pfaffmann, 2000).[5]

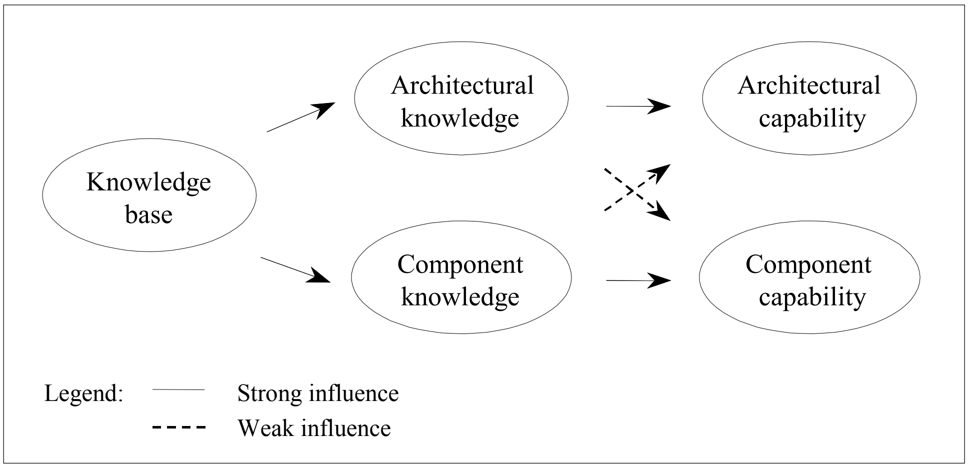
Conversely, *architectural capability* requires the coordination and integration of diverse bodies of substantive knowledge that are related to the *system behaviours* of components (Henderson and Cockburn, 1994; Pfaffmann, 2000). Such capabilities are essential in defining alternative product architectures and evaluating the degree of modularity of architectures, as well as other aspects of specific architectures, such as cost efficiency and delivered customer utility.

Architectural knowledge is the understanding of how to integrate the components which the integrating company does not necessarily have significant component-level knowledge about. Architectural capability is also based on knowledge about the applications and the 'use contexts' in which a product will be used (Pfaffmann, 1998a; 2000). Architectural capability often rests on *theoretical* knowledge about the system behaviours of the components that may be easier to codify and, hence, easier to transfer than knowledge about the component design (Pfaffmann and Scheuble, 1998).

Figure 5 illustrates the relationship between the knowledge bases of architectural capability and component capability. As can be seen, the distinction between architectural capability and component capability is not without some overlaps and intersections. Since architectural capability is necessary to craft feasible and modular product architectures, some degree of component knowledge (*i.e.*, a component's system behaviour) is essential. Likewise, the in-depth development of a component also requires an understanding of how the component will interact with other components and how the product it is in will be used.

Some degree of architectural knowledge (perhaps a great deal) should be maintained in-house, but this knowledge may be dispersed across many individuals. Thus, building, acting on and improving architectural knowledge is an inherently collective process. Moreover, important architectural knowledge may also lie outside the formal boundaries of business units or functional areas (Iansiti, 1995). Networked project structures and multifunctional teams are organisational measures in product development that aid the transfers of architectural knowledge. From a capabilities perspective, the project organisation of MCC in product development was intended to build architectural knowledge and capabilities, but did not achieve the necessary degree of architectural knowledge required to support the concurrent, independent development of systems in concept realisation.

Figure 5 The knowledge base and the related development capability



4.2 The assessment of architectural capabilities and component capabilities in product development

Architectural capability includes all the activities in product development to be carried out by an integrating company that defines and coordinates development for a specific final product, such as smart. Among others, these activities include the definition of the product architecture, the evaluation and integration of the components, the planning and implementation of the manufacturing processes and other specific tasks.

In a capability assessment, additional indicators can be evaluated and rated with a Likert scale (see Figure 6). Within this scale, values 1 and 2 represent low degrees, value 3, a moderate degree and values 4 and 5, high degrees of capability. The use of this kind of assessment tool reveals the tasks wherein a company possesses a profound understanding or has some weak spots. Such assessments can be carried out for each component in a product. Figure 6 illustrates a profile of architectural capability in the development of vehicle doors.

A similar procedure may be carried out to assess a company's component development capability for all the components considered important in a given development project. Figure 7 shows a profile of component development capability for a vehicle door.

This assessment covers both product development and manufacturing capabilities and, thus, can be used to detect the areas in need of capability building and improving.

Figure 6 The profile of architectural capability in the development of vehicle doors

Activities	Indicators	Tasks		Degree				
				Low		Moderate	High	
		No.		1	2	3	4	5
Definition of a specific product architecture	Degree of innovation of the architecture Project scope ...	A1	Identification of consumer needs				A1	
		A2	Transformation of needs into product functions					A2
		A3	Transformation into component structures					A3
		A4	Definition of specifications				A4	
		A5	Development of prototypes of the final product				A5	
		...						
Evaluation of components	Degree of innovation of components ...	B1	Evaluation of functional characteristics			B1		
		B2	Evaluation of customer utility		B2			
		B3	Estimation of target costs		B3			
		...	...					
Integration of components	Degree of innovation of components Internally or externally developed components	C1	Implementation of test programmes					C1
		C2	Optimisation of product maturity				C2	
		C3	Evaluation of component behaviour				C3	
		...	...					
							Score: 3.9	

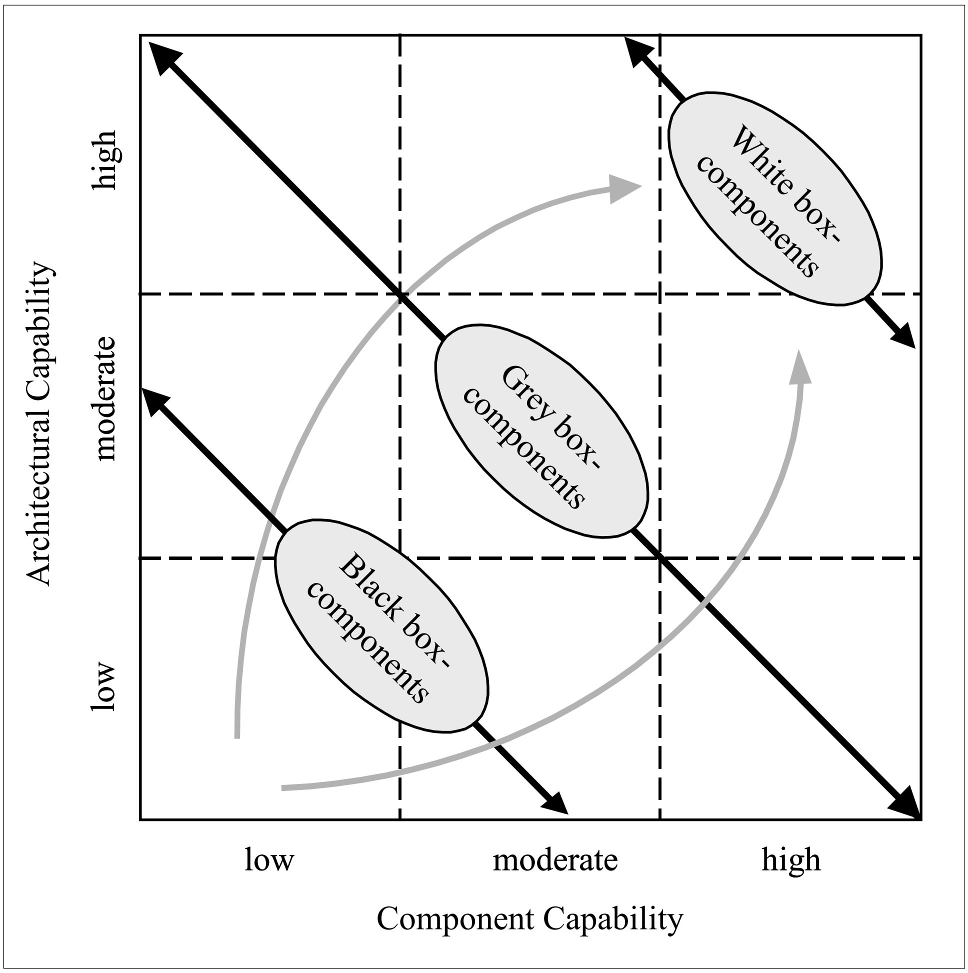
Figure 7 The component capability in the development of vehicle doors

Activities	Indicators	Tasks		Degree				
				Low		Moderate	High	
		No.		1	2	3	4	5
Concept development	Degree of innovation ...	A1	Development of concept studies			A1		
		A2	Crafting of prototype models			A2		
		A3	Detailing of component specifications				A3	
		...						
Concept realisa-tion of the component	Degree of innovation with respect to mass production ...	B1	Manufacutring of opti-mised prototypes		B1			
		B2	Evaluation of functi-onality and process feasibility of the component		B2			
		B3	Realisation of tests	B3				
		...						
Manufacturing of the component	Degree of innovation of employed process technologies ...	C1	Mass production of the component	C1				
		C2	Analysis of process costs			C2		
		C3	Quality control		C3			
		...						
							Score: 2.3	

4.3 A capability-based typology of components

Once the capability assessments for the components are completed, one can classify the components according to the degree of available in-house component capability and architectural capability. As an indicator of the stock of available knowledge in a firm, three levels of capability with respect to a firm's ability to develop components can be distinguished (*black*-box components, *gray*-box components and *white*-box components), as shown in Figure 8.[6]

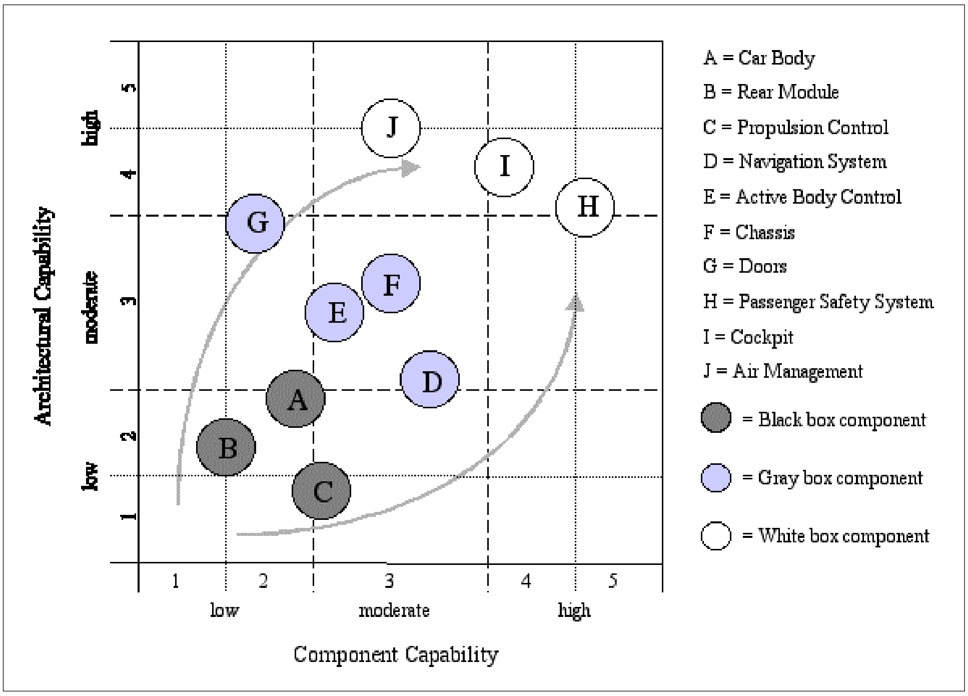
Figure 8 Capability-based typology of components



The information collected in the assessments of architectural and component capabilities can be used to represent a firm's capabilities with respect to specific components, as illustrated in Figure 9 (including the vehicle doors example).

The assessment and grouping of the component capabilities, shown in Figure 9, has important implications for outsourcing, early supplier integration and, thus, the modular organisation of product development.

Figure 9 Architectural and component capability-based evaluations of components (see online version for colours)



4.3.1 Black-box components

If an integrating firm's architectural and component capability is low – suggesting that the integrating company does not adequately understand the system behaviour of these components in the product architecture – the firm may not be certain that it can define adequate interface specifications. In this case (as in the case of MCC), the integrator may *describe* an architecture (Sanchez, 2001) of a preliminary nature with only the descriptions of the interfaces, rather than complete specifications. Furthermore, because the component capability is low, such components must usually be sourced from supplier firms. Working out full interface specifications for these components will, therefore, require an extremely close interaction between the integrator and supplier firms. Moreover, unless architectural capability can be increased during successive stages in the development process, outsourcing these components and achieving proper integration into the product will be a difficult and risky process. In this kind of situation, a search for alternative component types for which the integrating firm has greater architectural capabilities may be appropriate.

4.3.2 Gray-box components

The architectural and component capability is higher for gray-box components than for black-box components. In this case, early interface specifications may not be complete or fully correct and unforeseen problems in component interactions may be the result.

However, the architectural and component knowledge of the integrating firm may be sufficient to achieve an adequate level of integration in the concept development phase. The gray-box components in which a firm has moderate or higher component capabilities may give the firm the option to develop in-house components and/or to produce a given component. For instance, in the case of the navigation system (see Figure 9), MCC, as the integrating firm, possessed sufficient component knowledge to develop it in-house. The integrating firm may be able to achieve sufficient modularity in specifying the interfaces for this component to support modular organisation during concept realisation.

4.3.3 White-box components

If the architectural and component capability is high, the specifications should be appropriately defined in the concept development phase. The in-house development and production of components is possible, as well as outsourcing and supplier integration. If the degree of modularity specified for such components is sufficiently high, the modular organisation of product development should be possible without major difficulties.

5 Conclusion

This paper describes the approach of MCC to product development, outsourcing and supplier integration and the key role of modularity in the technical architecture of smart and its consequences for the organisational architecture of the development process. The high degree of *intended modularity* in the architecture of smart was a desired means to achieve several strategic objectives. First, more than 85% of the value added of smart was to be outsourced to supplier companies. Therefore, it was of prime importance that the development process be organised to make effective use of the specialised expertise and creativity of the suppliers. At the same time, the smooth integration of (supplier) components into the vehicle architecture could only be assured if the modular component interfaces were clearly specified in advance. Furthermore, direct interactions between MCC and the supplier companies were to be minimised in order to reduce organisational complexity during development and production. Thus, modularity was also the concept that would permit MCC to organise the development of a highly innovative vehicle with extremely tight resources. In the European automobile industry and especially for Mercedes-Benz, the modular approach of MCC was a pioneering effort.

However, MCC's approach to modularity was not without difficulties. MCC learned through experience that deficiencies in the degree of modularity in its product architecture led to deficiencies in the modularity of its organisation architecture. In fact, during the concept realisation phase of development, the architecture of smart was discovered to be insufficiently modular in supporting a truly modular development process. Insufficient modularisation was discovered in cross-sectional systems and modules and there was an inadequate specification of some functions and components. Although the effort to apply modularity was highly innovative, the company did not have the means to assess its in-house capabilities to define, specify and develop smart's architecture. Thus, the actual degree of technical modularity was not determined beforehand and the company overestimated the feasibility of its plans to develop smart through a modular organisation architecture.

To help overcome these difficulties, we outlined a tool for assessing a company's in-house architectural and component capabilities in product development. We suggested that although architectural and component capabilities do have some joint knowledge elements, most of the knowledge that forms the basis for the two kinds of capabilities is distinctive to each capability. We also suggested that different degrees of these capabilities imply different possibilities for outsourcing, supplier integration and the cooperative modular organisation of product development.

References

Baldwin, C.Y. and Clark, K.B. (1997) 'Managing in an age of modularity', *Harvard Business Review*, Vol. 75, September–October, pp.84–93.

Bensaou, B.M. and Pfaffmann, E. (1998) 'Mercedes-Benz and "Swatch": inventing the smart and the networked organization', INSEAD Case Study No. 07/98–4765, Fontainebleau.

Clark, K.B. and Fujimoto, T. (1991) *Product Development Performance: Strategy, Organization, and Management in the World Auto Industry*, Boston, MA: Harvard Business School Press.

Cooper, R.G. (1990) 'Stage-gate systems: a new tool for managing new products', *Business Horizons*, May–June, pp.44–54.

Cusumano, M.A. and Nobeoka, K. (1992) 'Strategy, structure and performance in product development: observations from the automobile industry', *Research Policy*, Vol. 21, pp.265–294.

Dosi, G. (1982) 'Technological paradigms and technological trajectories: a suggested interpretation of the determinants and directions of technical change', *Research Policy*, Vol. 11, pp.147–162.

Durand, T. (1992) 'Dual technological trees: assessing the intensity and strategic significance of technological change', *Research Policy*, Vol. 21, pp.361–380.

Fujimoto, T. (1999) 'Capability-building competition for lead time reduction: a case of front-loading problem solving in the auto industry', *Paper Presented on the Conference on R&D-Management 2000 – Challenges and Concepts*, Stuttgart, 23 July.

Fujimoto, T. and Yasumoto, M. (1998) 'The impact of product-industry characteristics on effective patterns of product development', Discussion Paper No. CIRJE-F-33, Department of Economics, University of Tokyo, Tokyo, December.

Galbraith, J.R. (1973) *Designing Complex Organizations*, Reading, MA: Addison-Wesley.

Göpfert, J. (1998) *Modularisierung in der Produktentwicklung: Ein Ansatz zur gemeinsamen Gestaltung von Technik und Organisation*, Wiesbaden: Gabler.

Gulati, R.K. and Eppinger, S.D. (1996) 'The coupling of product architecture and organizational structure decisions', Working Paper No. 3906, Cambridge, MA: Sloan School of Management, May.

Henderson, R.M. and Clark, K.B. (1990) 'Architectural innovation: the reconfiguration of existing product technologies and the failure of established firms', *Administrative Science Quarterly*, Vol. 35, pp.9–30.

Henderson, R. and Cockburn, I. (1994) 'Measuring competence? Exploring firm effects in pharmaceutical research', *Strategic Management Journal*, Vol. 15, pp.63–84.

Iansiti, M. (1995) 'Technology integration: managing technological evolution in a complex environment', *Research Policy*, Vol. 24, pp.521–542.

Iansiti, M. and Clark, K.B. (1994) 'Integration and dynamic capability: evidence from product development in automobiles and mainframe computers', *Journal of Industrial and Corporate Change*, Vol. 3, No. 2, pp.557–605.

Liker, J.K., Kamath, R., Wasti, R., Nazli, S. and Nagamachi, M. (1995) 'Integrating suppliers into fast-cycle product development', in J.K. Liker, J.E. Ettlie and J.C. Campbell (Eds.) *Engineered in Japan*, Oxford, New York, pp.152–191.

MacCormack, A., Verganti, R. and Iansiti, M. (1999) 'Developing products on "internet time": the anatomy of a flexible development process', Discussion paper, Harvard Business School, April.

Marples, D.L. (1961) 'The decisions of engineering design', *IEEE Transactions on Engineering Management*, Vol. 2, pp.55–71.

Nevins, J.L. and Whitney, D.E. (1989) *Concurrent Design of Products and Processes: A Strategy for the Next Generation in Manufacturing*, New York: McGraw-Hill.

Pfaffmann, E. (2000) 'Knowledge maturity of products, modularity, and the vertical boundaries of the firm', in N.J. Foss and V. Mahnke (Eds.) *New Directions in Strategy Research*, Oxford: Oxford University Press, pp.250–275.

Pfaffmann, E. (1998a) 'How does a product influence the boundaries of the firm? A theory of vertical integration and co-operation', *Paper Presented at the DRUID-Summer Conference*, Bornholm.

Pfaffmann, E. (1998b) 'Smart project?: co-operative supplier-integration in the development of a radical new city car concept: the case of micro compact car', Unpublished manuscript, Euro-Asia Centre, INSEAD, Fontainebleau, Center for International Management and Innovation, Hohenheim University, Stuttgart.

Pfaffmann, E. and Scheuble, S. (1998) 'Eine wissensbasierte Erklärung der vertikalen Grenzen einheitlicher Entscheidungshoheit', Diskussionspaper, Universität Hohenheim, LMU-München , Stuttgart, München.

Pfaffmann, E. and Stephan, M. (1999) 'Competence-based diversification in the world automotive supplier industry: some evidence from inward investment activities', *Paper Presented at the Workshop on Strategic Competence-Management*, Bochum, Germany, October.

Sanchez, R. (1998) 'Product and process architectures in the management of knowledge resources', in N.J. Foss and P.L. Robertson (Eds.) *Resources, Technology, and Strategy*, London: Routledge, pp.100–122.

Sanchez, R. (1999) 'Modular architectures in the marketing process', Special Issue, *Journal of Marketing*, Vol. 63, pp.92–111.

Sanchez, R. (2000) 'Modular architectures, knowledge assets, and organizational learning: new management processes for product creation', *Int. J. Technology Management*, Vol. 19, No. 6, pp.610–629.

Sanchez, R. (2001) 'Product, process, and knowledge architectures in organizational competence', in R. Sanchez (Ed.) *Knowledge Management and Organizational Competence*, Oxford: Oxford University Press, pp.227–250.

Sanchez, R. and Collins, R.P. (2001) 'Competing – and learning – in modular markets', *Long Range Planning*, Vol. 34, No. 6, pp.645–667.

Sanchez, R. and Mahoney, J.T. (1996) 'Modularity, flexibility, and knowledge management in product and organization design', *Strategic Management Journal*, Vol. 17, Winter Special Issue, pp.63–76.

Sanchez, R. and Mahoney, J.T. (2001) 'Modularity and dynamic capabilities', in T. Elfring and H. Volberda (Eds.) *Rethinking Strategy*, Thousand Oaks, CA: Sage Publications, pp.158–171.

Thomke, S. and Fujimoto, T. (1999) 'The effect of "front-loading" problem-solving on product development performance', Working Paper 98-103, Harvard Business School, Cambridge, Massachusetts (revised May 1999).

Thomke, S.H. (1998) 'Simulation, learning and R&D performance: evidence from automotive development', *Research Policy*, Vol. 27, pp.55–74.

Thomke, S.H., von Hippel, E.A. and Franke, R.R. (1997) 'Modes of experimentation: an innovation process – and competitive – variable', Working Paper No. 97-052, Harvard Business School, Cambridge, Massachusetts.

Ulrich, K. and Tung, K. (1991) 'Fundamentals of product modularity', *Issues in Design Manufacture/Integration, ASME, DE*, Vol. 39, pp.73–79.

Ulrich, K.T. (1995) 'The role of product architecture in the manufacturing firm', *Research Policy*, Vol. 24, pp.419–440.

Ulrich, K.T. and Eppinger, S.D. (1995) *Product Design and Development*, New York: McGraw-Hill.

Wheelwright, S.C. and Clark, K.B. (1992) *Revolutionizing Product Development – Quantum Leaps in Speed, Efficiency, and Quality*, New York.

Whitney, D.E. (1993) 'Integrated design and manufacturing in Japan', *Manufacturing Review*, Vol. 6, No. 4, pp.329–342.

Notes

1 See Baldwin and Clark (1997) for the concept of 'modularity in use'.

2 The literature on product development processes is characterised by a rich diversity of approaches and insights. See, for instance, Marples (1961), Galbraith (1973), Nevins and Whitney (1989), Cooper (1990), Clark and Fujimoto (1991), Cusumano and Nobeoka (1992), Wheelwright and Clark (1992), Iansiti and Clark (1994), Liker *et al.* (1995), Ulrich and Eppinger (1995), Thomke *et al.* (1997), Fujimoto and Yasumoto (1998), Thomke (1998), MacCormack *et al.* (1999), Fujimoto (1999), Thomke and Fujimoto (1999) and Sanchez (2000).

3 For the importance of product architectures in product development, see Ulrich and Tung (1991), Ulrich (1995), Ulrich and Eppinger (1995), Gulati and Eppinger (1996), Sanchez and Mahoney (1996), Göpfert (1998), Sanchez (1998), Pfaffmann and Stephan (1999), Pfaffmann (2000), Sanchez (1999) and Sanchez (2000).

4 In the following discussion, all the constituent functional parts of a product will be called 'components', regardless of whether they are modules, systems, system-modules, components or parts.

5 This is not the case in all organisations, however. For a case study of GE-Fanuc Automation, in which both component and architectural knowledge and design methodologies are systematically articulated and codified, see Sanchez and Collins (2001).

6 These component classes reflect the existing stock of knowledge of a given firm, not the inherent physical properties of various components. Thus, a component can be a black box for firm X, but a white box for firm Z.