ARCHITECTS IN NEW PRODUCT DEVELOPMENT – A NEW WAY TO DERIVE SYNERGIES BETWEEN PROJECTS OR BRANDS IN CAR INDUSTRY?

11th GERPISA International Colloquium

Jon MIKAELSSON, Fenix, Chalmers University of Technology

Blanche SEGRESTIN, Ecole des Mines de Paris,

Benoît WEIL, Ecole des Mines de Paris,

Jon.mikaelsson@fenix.chalmers.se, Blanche.Segrestin@ensmp.fr, Bweil@ensmp.fr

In recent years, there has been a growing concern in the automotive sector on product development performance, with an increase of product variants, a decrease of products' life-cycles and a shift towards various services development. Internal growth, mergers and acquisitions have also reshaped the sector. As a result the global and multiple brands perspective plays a more prominent role, where each brand has to differentiate itself to form a profitable independent business and at the same time needs to take advantage of strategic common orientations and cross-brands synergies. A key issue is therefore to detect new leverages to organise these synergies [1].

With regards to these strategic evolutions, new technical and organisational solutions have been introduced, such as platforms development, modularisation and full service suppliers. Some authors have shown that the implementation of such solutions have considerably improved product development performance [2-4]. Most manufacturers also seek to enhance cross-brand cooperation with them. Yet, the implementation of these solutions encounters important difficulties [5]. Platforms are for instance not as long-term stable as expected making it difficult to achieve and maintain a high commonality ratio (i.e. the number of components or costs shared by different vehicles), the results are somehow disappointing [6,7]. M. Sako has also showed that targets of outsourcing thanks to modularization were probably "neither feasible, no desirable" [8]. Manufacturers are therefore trying to find new ways to take advantage from multi-projects management and cooperation between projects or between brands.

This paper aims at addressing this overall issue and more especially the following question: what are the principles to design the product so that synergies are possible? The paper is based on the analysis of the case of a European car manufacturer. The latter has engaged a strategic reflection and one of the authors of this paper is directly involved and actively participates in the related managerial processes.

Given the limits of traditional managerial doctrines on inter-projects synergies, this approach is an attempt to characterise the design process of architecture, i.e. the higher level of product definition, from which modules and systems can be specified. Although very recently undertaken, this approach has appeared promising both from a managerial perspective and from a theoretical point of view: it helps us, as a starting point, to discuss the notion of architectural design.

This paper is organised as follow:

- In the first section, we examine the limits of traditional approaches in the automotive sector. Platforms' development or modularity are difficult to implement, because they are based on physical components, technologies or interfaces that are supposed to be stabilised for reuse in different projects. Looking more closely to platforms and modules, the perimeter and the nature of interfaces appear rather ill defined.
- In the second section, we show that these managerial doctrines refer to the notion of architecture. Although widespread, this notion is rarely considered as something to be designed, and speaking about "design rules", we miss how to design them: what is at stake is to capture how an architecture can be designed. In other words, what are the role and the competence of an architect?
- In the third section, we will present the case of a new approach that is currently developed by an European manufacturer and which seems promising. We will try to capture its logic and principles. A global and/or cross brand car manufacturing system is a complex one. Considering the unavoidable amount of interdependencies in such a complex system, one needs to focus on a higher design level to understand how technologies, subsystems or modules interact. We argue that such an approach suggests an interesting representation of the problem, but we will also discuss its limits.

Hence, this paper is a preliminary effort and a starting point of a research program to study the design rules of architecture: how to characterise a good architecture design? We will argue that the ability to derive synergies across brands and projects without decreasing their innovativeness and uniqueness requires specific principles. When apprehended as an object to be designed up-front, in a very complex context, the product architecture may lead to a deep renewal of managerial doctrines. We will discuss the ways to reframe architecture, and the principles that could structure the architectural design. Based on on-going reflections, several important research questions can then be presented.

I- SYNERGIES ACROSS PROJECTS: THE PHANTOM OF ARCHITECTURE

With the acceleration in product renewal rates, organisations' way to focus on successive projects is being heavily criticised. Projects are being accused of wasting resources on solving recurrent problems and not gaining sufficient benefit from past experiences and investments. By focusing on different forms of synergies between the models, the aim now is to counterbalance this single project focus (the "fat projects" trend [4]), and so facilitating multiply product variations with less resources and with less project complexity, leading to shorter lead-times and higher quality.

How to derive synergies among projects? This question is highly problematic. Although much has been said on knowledge transfer, apart from the time constraints and the difficulties to memorise and communicate highly specialised competencies, the problem is that not all knowledge is helpful for all projects. Projects are always highly contextual and knowledge is seldom of a plug-and-play kind but asks to be reframed for each project to fit with local compromises [9,10]. It has been shown that functional expertise needed to be strengthened to build up successful projects [11].

Another approach consists in developing technologies or components that embed knowledge to the benefit of further projects. The development of common platform and modules is probably today the most widespread endeavour in this perspective.

A- Platforms and modules: general assumptions

According to Cusumano and Nobeoka, in the automobile industry this type of coordination involves the transfer of the "platform" from one project to another. Drawing on a comparative analysis, the authors demonstrate the effectiveness of the notion of simultaneous transfer. They show that the introduction of a complete new concept ask for longer lead-time and more development work as the projects need to build each components from the beginning in an innovative way. In parallel, when a project is based on sequential transfer of technologies or on an existing concept, it is necessary to spend time to test and validate existing technologies in the new project. On the contrary, the projects that take advantage from simultaneous transfer need decreased lead-time as they allow both a job sharing among projects and real-time adjustments and interactions.

Under these conditions, there is also renewed interest for the concept of modularity. Modular management is very popular in the computer industry, and seeks in practice to fix the architecture of a product and the interfaces between the components very early on in the design process. In this way, the different components can be developed and renewed in parallel. This implies that the manufacturer must have excellent knowledge of the potential interdependencies between subsystems to be able to decentralise the development of complete modules [2,3]. Among the main advantages, the most frequently put forward are the following:

- By isolating modules that are transparent for the customers, it is possible to reuse, add or substitute some of them in different products, and therefore reduce investments.
- Modularity is also said to improve flexibility in production and outsourcing: it is said to reduce lead-time and to improve milestones predictability. With an appropriate partitioning of the product, it may possible to select suppliers that are capable and have developed a specialised expertise in a limited field.
- Furthermore, modularity is said to enable a high rate of innovation on component technologies and increasing differentiation. If modules are loosely coupled to each other, then it may be possible to accelerate variations on only some of them. By reusing main modules, the idea is to develop rapidly and at low costs some variations to test the customers preferences and therefore develop products that fit better with the market demands and its evolutions [2,12].
- Sanchez has also argued that modularity has also implications on knowledge management: “an early definition of process capability requirements enables concurrent development of both new product components and the new process capabilities needed to make and assemble new components”. More generally, a modular architecture would help to “know what we know” and to identify “hidden capability bottlenecks” to leverage skills and competencies [13].

B- Limits

Practically, the organizational solutions combine matrix organisations, involving dual responsibility for engineers to improve information sharing, and development centres responsible for development of common solutions to several projects. With modularization, the organizational structure is assumed to mirror the product decomposition: As noted by Sanchez, the modularity of the products reflects a mode of coordination ("embedded coordination") that is particularly effective [14]. However, it has been shown that this consistency is not self-evident and the organizational inertia is likely on the contrary to hinder the change in product architecture, especially since the modularity may have different purposes ("modularity in design, in use or in production) [8].

Besides, and probably more importantly, different empirical studies have shown us that platforms and modules were based on assumptions that are not generally fulfilled:

Standardization versus the need of contextual compromises

It should be noted that to implement this strategy, at a product level, specific modules or technologies must be separated out from the rest of the vehicle so that they can be integrated in different vehicles. Given the levels of cross-functional performance and the interdependency of the different components, this "standardisation" of a specific function, or even an entire wheel base in the case of platforms, is only possible if each of the vehicle's sub-units can be made to be relatively independent from the rest of the vehicle.

Yet, each car is a highly complex system that requires huge amount of final tuning: cars result from very contingent compromises, that are highly specific and contextual [15,16], between a huge number of parameters (typically 10.000 components, each with an associated network of manufacturing process, enabling 100 complete vehicle variants per main design, each with 6000 defined specifications to offer 100 functions, in order to capture 100 high level customer and legislation demands).

Problematic interactions between the different sub-systems

In these conditions, it is difficult to anticipate and to depict in advance the problematic interactions [17], especially in a context of innovation: it is unrealistic to test each interaction and the validation tools need to focus on a small number of interactions that are likely to cause problems. But as any innovation might activate new interactions in an unexpected way, it doubtful that interfaces can be robust in the development process.

By shifting from the idea of shared solutions (technologies or physical components) to the idea of common interfaces, contextual adjustments on different components are assumed possible to fit with the specific features of each car. In computer industry for instance, the interfaces are sometimes described as a functional components: a buffer layer that isolates the microprocessor from the rest of the system is one example at Intel [18]. But still, the problem is the same: how should an interface be standardised? How to specify it in order to avoid interactions between the numerous subsystems? It is difficult too see how an interface could be fully specified without interfering with the surrounding components. On the contrary, we can think that a fully specified interface would determine most of the parameters of the surrounding components… That is again the problem of interdependencies and transversal functions that impact every component and need unstable compromises [9]. Therefore, the idea of decoupling different subsystems appears quite theoretical when going back to automotive industry [17].

Specifications are not known in advance

Furthermore, specifications are not known in advance: first, within a car, there are different logics and cycles. Developing the wagon, we refer to “static” structure mechanics that are very styling intensive and that relate to manufacturing technologies such as shaping of metal sheets. Conversely, developing power-train, we refer to the area of thermodynamics and dynamic mechanics, and a dominant manufacturing technology is casting. Testing and manufacturing of these two dominant domains are different, making a combined effort to change both in parallel very difficult from a management perspective.

Secondly, platform life-cycle planning becomes an act spanning over decades. While the lead-time (including planning) for a project is between 3-5 years, platforms (that relate to several but seldom complete parallel project has life-cycles of 8 to 12 years. Thus, planning just two cycles ahead create a need to consider twenty years ahead. Indeed a very long period in a technology intensive branch.

Furthermore, attributes and requirements evolve constantly and they are not given but need to be designed. Even when the attributes are defined, distributing a series of objectives among the different sub-units is an extremely delicate exercise, usually carried out by a cross-disciplinary expert. Cross-disciplinary experts are therefore widely responsible for the overall project coherency. They must not fix objectives that are too low, otherwise the overall objectives will not be reached, or too high, to minimise development constraints. Their work must take interdependent factors into account, which, as we have seen, are often very badly known. Hence, to deploy objectives is a complex task that takes time and asks for detailed and continuous adjustments. Balancing objectives and solutions due to the result of testing at various system levels is then a both time-consuming and complex task.

In these conditions, to stabilise technologies, components or interfaces appear extremely difficult as the functional requirements themselves are not known and their interactions quite impossible to predict.

II- SUPPORTING INNOVATION BY STABILISED ARCHITECTURE?

At this stage, we've observed that the scope of a platform or a module is somehow ill-defined: a platform doesn't constitute a natural subsystem; what can be commonized or not has to be carefully studied and some unexpected problems of feasibility can appear very late in the development processes. M. Sako has also noticed that modules have different boundaries from model to model [8]. Despite the well-known concepts of platforms and modules, the notions are quite confused and uncertain when looking more closely at their significance.

Then, platforms and interfaces are not natural objects. They have to be planned, designed and carefully specified. But what are the underlying principles? What are the conditions to stabilise interfaces? The literature has often stressed on the advantages and the difficulties of platforms strategies or modularization. But, from a managerial point of view, one can wonder what are the conditions and leverages to actually develop standardised interfaces or common components. How to define the scope of a platform? How to select the interfaces to be frozen and how to define the scope of modules?

An underlying layer: the product architecture

Interestingly, when we refer to literature, modularity and platforms are always related to the architecture of the product. In a common sense, architecture relates to packaging and volume allocation: Usually, the role of architects is to propose some allocations of volumes to the engineering departments and the experts from each cross-disciplinary function (acoustics, endurance, etc.). But more importantly, architecture concerns the way different subsystems are related to each other [19]. In this respect, both platform and modules refer to architecture in different ways:

- Platforms are considered by Cusumano and Nobeoka as "a key-technology" that are basically overlapping the underbody frame (floor pan, suspension system...). As they explain, "it is mainly the *architecture* of the product, even if a modern car is made up of more than 30 000 components. It structures the main systems of he vehicle as it determines the size of the body, the size and type of the engine and of the transmission system" [4]. Besides, it is generally the most expensive subsystem both in terms of development and manufacturing activities, with very critical know-how.

- In the computer industry, Intel is also seen as an architect as it succeeded in defining and stabilizing the system around the microprocessor with open and stabilised interfaces [18]. As for the modularity, the main concept here refers to the ability to define clear interfaces between components or systems in order to allow both parallel renewals (differentiation and innovations), and compatibility and integration of multiple functions.

- Sanchez gives a more precise definition: “an *architecture* is a system design for which designers have specified

    1) the way the overall functionalities of the product or process design are decomposed into individual functional components and

    2) the ways in which the individual functional components interact to provide the overall functionalities of the system design. Component interactions generally are described by component interface specifications that define the inputs and the outputs that cross the interfaces between interacting components.”

Ulrich has a quite similar definition of modularity when he writes: "A modular architecture includes one-to-one mapping from functions elements in the functional structure to the physical components of the product, and specified de-coupled interfaces between components" [20]. Consistently with former work [21,22], these authors see the architecture of a product as the bundles between its functionalities (the attributes for the customer) and the technical detailed solutions.

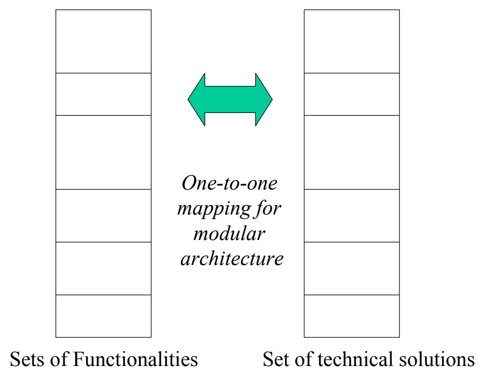

Figure 1: The theoretical modular architecture

Other definitions are consistent with these ones: for Fujimoto, "Generally speaking, when we say product architecture, we are talking about a basic design concept for the product, that is, how we split the product into modules, then, how to allocate functions to each module, and how to design and coordinate interfaces (a linkage through which information and energy are taken in and out) among different parts, or modules" [23]. For Baldwin and Clark, "an architecture [is a category of design rules], which specifies what modules will be part of the system and what their functions will be" [3].

Architecture as a "dominant design": a realistic approach for complex fields

These definitions are interesting, but what are their managerial implications? One can notice that they consider architecture as something that is already established, as a characteristic of a product (or a process, or a service), or as some rules that need to be specified in advance. The approaches presented former actually consider the architecture either as a result of a historical path, and therefore the architecture is quite synonymous to a "dominant design" that would be embedded into the development processes, or as a result of former design work.

According to [24], products innovations follow a process of different steps: at the first step, many technologies and targets compete, but then the rationalization eliminates most of approaches and makes a "dominant design" emerge. At the end, the product is considered as mature and the innovation concerns mainly processes and the competition is based on very few variables (price and quality on identified criteria). Today, the car can be considered as a mature product, and it can be an objective to maintain this dominant design:

In such a complex field, it is necessary to preserve as much as possible the existing solutions when designing a new car: It is too costly and too risky to introduce too many innovations and even very incremental changes can destabilise earlier compromises in an unexpected way. In this perspective, the architecture usually appears as the dominant design (that is transferred from one car to the following ones): This strategy has several managerial interests. With (control or management) of architecture development, you are able to stabilize over time interfaces between sub-systems; it enables:

- Coordination between engineers that can work in parallel, but autonomously, on different systems,
- Innovation on one system without changing the systems around,
- Delegation of the development of complete systems or modules to external suppliers or partners.
- Focus the need for innovation (customer perspective) to areas with low risk/low cost/low complexity

So, most of the times, it seems efficient. Here, we can notice that platform concept, modularity and the emerging approach are all referring to a Dominant Design. The architecture is considered as a result of a rationalization process that has progressively stabilised all major systems, functions and interfaces. For other authors, as Baldwin and Clark, architecture results from the design work, but how are the design rules defined?

The literature obviously urges for a renewal: it doesn't address the role of the architect and this could impede architectural innovation. Therefore, an important issue consists in capturing the role of architects who design effectively the product architecture: what is at stake is to depict the efficient principles of architectural design that enable designers to derive synergies between projects.

It will be the purpose of the next section. Based on an empirical on-going work, we try to apprehend architecture not only as an ex post characteristics of a product, but as an object to be designed.

III- HOW TO DESIGN ARCHITECTURE? THE CASE OF A EUROPEAN MANUFACTURER

To apprehend these issues, many manufacturers have experimented with various strategies and lots of investigations have been done. We had the opportunity to examine/follow a new approach in one European manufacturer. Although emerging, this approach tends to be formalised and discussed within the company and it was a good vehicle to analyse the problem both from a theoretical and managerial point of view. In this section, we briefly present this approach, its promising features and its limits.

Capturing the roots of instability

Considering that the main hinders for modules or platforms were due to the numerous and unexpected interdependencies, the issue was to depict the roots of instability: what generates modifications? Why is it necessary to modify tools and components despite of the willingness to derive synergies across projects? What are the main factors of instability?

A thorough analysis has shown that some critical thresholds of requirements were responsible for the most costly changes. It appeared that the main additional costs to develop a new model on a common platform were due to the necessity to modify some key-parameters of the parameters. In fact, the costs of platforms changes reveal themselves to be highly correlated with the changes in few alternatives: for instance, if the location of the battery is changed, if some changes are required in the type of suspension, or in the type of engine fixation, then, there is a very high probability that the platform can not be stable anymore: most important tools or systems would require change. Another example refers to the electrical system: the hardware/software ratio determines the electrical configuration and if some changes are required, then, all related systems and components may change to fulfil the mechanical/electrical requirements.

Hence, the additional costs or the instability can be explained by “intermediate parameters”, that are neither functional dimensions, nor technical solutions. The assembler has gathered these “intermediate” critical dimensions into five main “architectural objects”, namely around power-train, body, chassis, climate and electrical system. What is interesting in this approach is that it contributes to build a language that can be used as an intermediate between different languages: the manufacturing languages, the attributes (customer-oriented functions), and the engineering design parameters. Besides, the intermediate parameters are not numerous: what is interesting is that if these few critical dimensions are stable, that it may be possible to achieve good commonality ratio, or more precisely: when these critical dimensions cannot be common between different models, then, there is very little room for platform sharing.

The following figure shows how the underlying idea consists in defining a high level of architectural dimensions as an enabler of platform definition, from which a stream of derivative models can be efficiently developed.

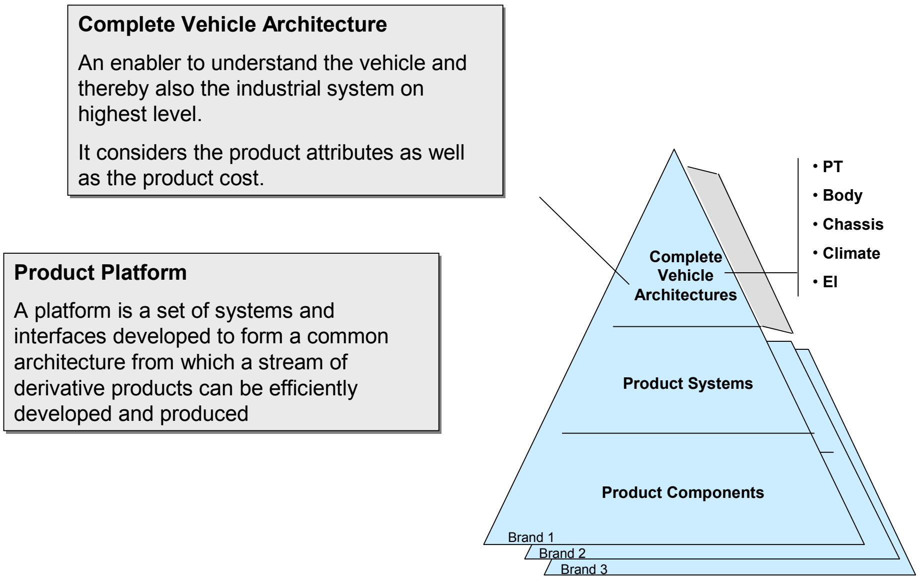

figure 2: vehicle architecture and platform

From common elements to "half-designed products"

A first result concerns the level of analysis. It shows that a quite limited numbers of items could explain the main part of instability. A too detailed level of analysis would probably have been difficult to cope with. By detecting the major dimensions that impede commonalization, this analysis provide an actionable tool. It becomes possible to evaluate in early stages, when specifying a new vehicle, whether it will fit or not with the existing platform (and/or existing and dominant designs and technologies).

The analysis doesn't try to capture all the interdependencies within the very intricate system of a car. Given the very high level of interrelations, a new attribute or a specific functionality is likely to imply many changes. But the attempt here is only to map the main dimensions that characterise a common architecture to enable common platform.

As a matter of fact, the suggestion was here to consider these critical dimensions as the overall structure that constitutes the restrains and capabilities within a platform: no modifications should be allowed on this level in order to achieve a satisfying level of commonality. Maintaining these generic dimensions is, in a sense, a prerequisite to develop valuable platforms for several different projects.

In our sense, this approach avoids some important pitfalls:

- It doesn't refer to detailed solutions but to conceptual level and therefore it allows contextual compromises and tuning on each model.
- It doesn't assume that there can be some decoupling between different subsystems. It doesn't seek to identify the interfaces that could be frozen: on the contrary, it aims at identifying the major interactions that generates extra costs.

- It doesn't seek to achieve a complete picture of the system, but only to capture the most critical dimensions. In other words, it results into a limited numbers of parameters that managers can handle.

At this stage, the critical dimensions could probably be enough to specify the notion of platform. It has been suggested to call the list of dimensions "the complete vehicle architecture". As a matter of fact, these dimensions characterise the specificities of a platform and provide a framework to depict whether or not the platform is still the same or if it has been modified into a new platform.

In this respect, the concept of platform can be formalised in a more satisfying way. Seen as a set of critical dimensions, the platform is no more a series of fully specified parts. What is interesting in this proposal is that it refers neither to detailed solutions nor to physical interfaces. Here, the concept of platform has deeply changed to the benefit of the notion introduced by Benoît Weil, the notion of half-designed product. It is neither a set of components nor a set of interfaces. It has much to do with the structuring dimensions of the car. In other words, it relies on a representation of a “half-designed product” [9]: some dimensions are defined as common for a set of cars, but these dimensions are not physical solutions. The half-designed product is a series of pre-validated parameters; it is only half-designed because the onus is on the successive projects to fine-tune the initial series of parameters to suit a given set of values and to choose the other action parameters depending on their constraints and objectives.

The concrete solutions have still to be defined depending on each car’s specificities. Hence, it is expected that a wide range of choices is possible to differentiate the different cars, but that these common dimensions enable to preserve a good level of interoperability for some tools, and a good level of commonality for some components (these tools and these components being presumed to induce the most important part of costs and quality).

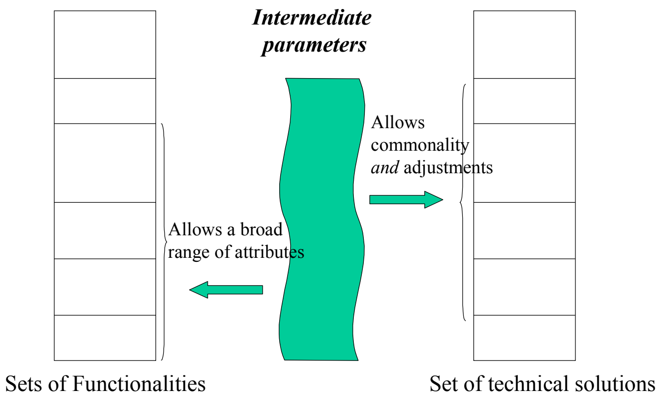

Figure 3: capturing the intermediate dimensions

A powerful managerial devices?

In this section, we discuss first the potentials of this approach and secondly its possible limits. It leads us, thirdly to suggest how to measure an architecture's performances and therefore to guide designers activities.

Potentials: early dialog and front-loading

With this new approach, it may be possible to evaluate to what extent a new specification would imply some revisions in tooling and development process, and the induced costs of it. The same is also valid concerning a change in technology.

In other words, defining attributes (and technologies) for a new car, people in charge of the definition of attributes (respectively of technologies) could see whether the attribute (resp. technologies) fit with platform definition. They could be aware of the induced costs and it becomes possible to make them responsible for the budgets. It is nevertheless important to develop ways of simulating and evaluating in early phases the different alternatives for all the parameters concerned. In this respect, these intermediate dimensions are a powerful devices for front-loading problem solving [25].

This creates a sort of language around which it may be also possible to organize the dialog and the coordination between the various actors of the development process. The half-designed product is a tool for coordination between the designers in the common sub-unit and the units specific to each vehicle. This represents a radical change in the type of interaction: cooperation no longer consists in negotiations between the project teams to define the specifications for the different parts, independently from any constraints arising from these specifications further down the line, but rather a joint learning process concerning the interactions between the design parameters and the functional targets, work designed to evaluate the impact of a choice on each vehicle.

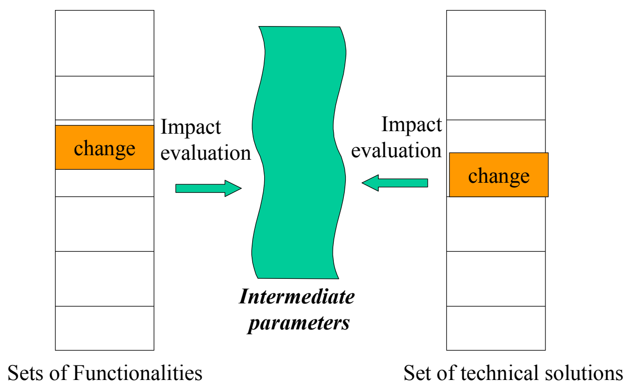

Figure 4: Evaluation means of potential impact

Once intermediate dimensions are frozen, which flexibility?

By characterising architectural design in this way, we do not mean that one should put on the table all the dominant design in automotive industry! We do not mean that the design process is to be started again from the beginning! It is actually quite impossible or unrealistic to reconsider all functions and build consistent building blocks that are loosely coupled…

But this way to apprehend architecture may stimulate new approaches. As a matter of fact, the investigation we have mentioned is a first attempt to depict the interdependencies.

Then, the challenge can be sum up in the following ways: We can wonder if the architectural design is not likely to prevent “architectural innovation”. As a matter of fact, once the intermediate dimensions are defined, they constrain the attributes scope. Then, one can fear that some major innovations would be impossible to introduce. Besides, why should these dimensions be the more relevant ones in 5 or 10 years? Is it so that the value of architecture in commonality related choices is valuable only when technology is stable over long time (15-20 years)?

To design efficiently architecture, the question is: what would be the requirements for an architectural design to allow innovation? Probably it mustn’t be completely frozen: what means freezing these dimensions? Is it realistic to impede any modification on these dimensions? And what would be the consequences of that?

Furthermore, the reasoning is possible only once the platform is defined. At this stage, it is an *afterwards* evaluation. Would it be possible to evaluate *in advance* the range of possible choices (attributes choices, technological choices) or the scope of what is impeded? If this evaluation was possible, then it could provide a criterion to judge whether we have a ‘good’ or ‘bad’ platform design: it would be bad if the scope of possible choices is too narrow…

Then, we see that the question is to design a platform (i.e. to specify some critical dimensions) so that it doesn’t reduce the scope of possible choices. Our hypothesis is that it refers not only to the specification of the platform (frozen and stable dimensions) but more generally to the architectural design: the specification of the architecture being the split of functionalities into soft and robust parameters.

Architecture performances: which measurement?

In these conditions, it seems important to reformulate more clearly the notion of architecture. Architecture is not only a result from a historical development process. It is not necessarily defined in advance. It may be something to be designed (the role of an architect) in order to achieve an overall functional capacity to provide the targeted service to the end-user. Designing architecture rather than observe it afterwards is moreover important once it is acknowledged that not all architectures enable commonality *and* differentiation.

Going back to the definition of Sanchez or of Baldwin and Clark, it refers to the design work, where the overall functionalities are identified and where they are split into a list of various functions to be achieved and provided by different subsystems[1].

In a sense, the architecture could refer to the way the overall attributes of a vehicle could be split into two decoupled sets of functions, loosely coupled to each other. But what the former discussion has revealed is that, to enable synergies *and* differentiation, the architecture may require some specific features: it must for instance identify a set of "stable functions" and a set of differentiated functions for which the scope of variation must be quite broad. (this prerequisite is perhaps already fulfilled, regarding the existing overall dominant design logic of a car, and perhaps not likely to change rapidly due to the enormous complexity regarding interfaces between “the car” and the social and physical infrastructure.

[1] It is closed to the Systematic design Theory and its first step of design process: this theoretical model of design activities, activities are divided in different levels : a functional definition, a conceptual level, where the principle to achieve a certain function is selected; and the embodiment level, where the components are designed in a detailed way [26]

Perhaps the problem of a platform could be seen in the following way: which parameters can be set in principle without significantly reducing the designers' level of freedom and also maintaining a wide enough scope for performance and design? The idea would be to depict a set of parameters/functions that can be stabilised and a set of function that needs adjustments and differentiation. What is at stake is *to reformulate the functional spaces* to identify, instead of the traditional specifications, a set of *robust* parameters (that can resist to attributes variations) *and* a set of *soft* parameters (that enable differentiation and uniqueness). It is worth noting that the soft parameters can be partly identified, but can also be partly unpredictable.

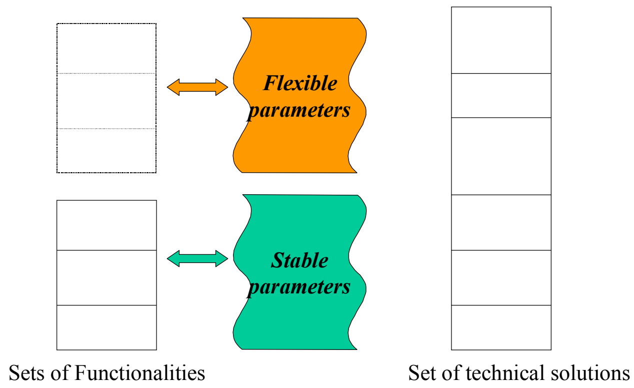

Figure 5: distinguishing stable and flexible dimensions

Of course, at this point of time, we can not provide any concrete solutions about how to define and implement such a definition. But this approach gives some clues about the method to apprehend it:

Implications from the stable sets of parameter: niches and clustering

The problem is not only to identify the critical dimensions that may be responsible of instability. The problem is rather: if we freeze these dimensions, what is the scope of possible choices? The challenge is therefore to evaluate the scope of possibilities within the constraints. Is it possible? How? How to evaluate if the frozen or stable parameters are too constraining?

We can see that the architectural design could become a highly powerful or normative tool to set up a new language for platforms design within a company or within a group of different brands.

There is no reason, for instance, to believe that the clustering principles that gather different vehicles around the same platform are robust in the new framework. What would it imply to consider a cluster of vehicles that have in common the engine position, or the battery location?

Furthermore, if the critical dimensions are actually captured to enable effective economies of scale, then, this new approach would help to identify "what is possible to differentiate" on the basis of this architecture. At low costs, it would then be possible to generate multiple variations and to identify valuable niches. If it is so, we can expect that architectural design would renew the offers from manufacturers.

Implications from the flexible sets of parameter: coordinating collective learning process

Then, another challenge is to identify what are the parameters that need to be differentiated or that will probably change in the coming years? Where are the "mutant genes"? New investigations are necessary, and these investigations probably require new methods and new tools to explicit the set of differentiation: can it be done by simulation?

In terms of coordination, the half-designed product concept is enlightening as it "in fact refers to the state of knowledge accumulated on a project at the interface between designer and user (or person to whom it is addressed): on the one hand, the half-designed product includes the stabilised knowledge by identifying the parameters that must not be touched; on the other, it points out the parameters on which further work must be done" [27]. Here appears another very interesting potential contribution of the new approach. The architectural design doesn't seek to impose a set of constraints to people in charge of defining attributes or to the engineering departments. It is obviously not only a tool to evaluate attributes or technologies' choices: it can be seen as a tool to guide engineering and marketing exploration and learning effort. Today, engineering departments can be criticised because they often defend technological innovations for the technology *per se*. But, if some emerging attributes don't fit with the present critical dimensions, then it means that some "lockers" are identified, and that they require further learning (cf. the "hidden capability bottlenecks" mentioned by Sanchez).

Hence, technological innovation could probably be stimulated toward some focused and specific directions. Knowledge production can then be "polarised" and "organised". Hence, the notion of architectural design is useful as a managerial tool to involve all the stakeholders and to lead them to work on common questions.

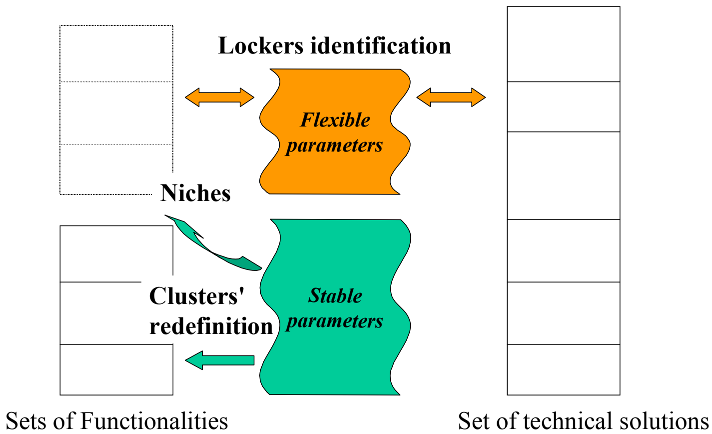

Figure 6: further implications

CONCLUSION

In this paper, we have tried to set up the basement for a coming research program. We analysed an emerging reflection to design platform and cars with enabling synergies across projects. By examining the potentials and the predictable limits of this reflection, we've tried to clarify the notion of architecture. This has lead us to question the underlying assumptions of current managerial doctrines on platform and modules. The latter generally refer to a given dominant design. The architecture is however not a given and stabilised set of parameters. To efficiently derive some synergies without narrowing the scope of differentiation, the architecture has to meet certain requirements. It is therefore necessary to investigate which are the parameters that need to be stabilised, and what it does imply. We have then suggested, on the basis of on-going reflection, some methodological elements to apprehend architecture and architectural design.

Given the on-going process, this paper is mainly introductory: it aims at addressing some issues for further research. Among these issues, we can mention the following:

- Following research on innovation management, it would be necessary to lead further investigation on the actual ways of reasoning of architects to capture the conditions of implementation of the new approach: what would be the necessary competencies to develop? How to stimulate learning processes to reframe "functional spaces" in more consistent ways? How to investigate on mutant functions and the subsequent technological lockers?
- How, from a managerial perspective, develop some decision making principles to cope with the trade-off between product singularity and synergies?
- What would be the impact of such competencies on the current matrix structures and on the principles of project development? What would be the impact of the new approach on the intricate network of firms in the automotive sector? How would the role of suppliers and partners influenced? To what extent would the relationships within a multi-brands group change?

[1] Ulrich Jürgens, *New Product Development and Production Networks,* Springer, 2000, 472 p.

[2] Carliss Y. Baldwin and Kim B. Clark, *Design Rules, volume One : the power of Modularity,* MIT Press, 2000.

[3] Carliss Y. Baldwin and Kim B. Clark, "Managing in an age of Modularity", *Harvard Business Review,* September-October, 1997.

[4] Michael Cusumano and Kentaro Nobeoka, *Thinking beyond lean, how multiproject management is transforming product development at Toyota and other companies,* The Free Press, 1998.

[5] Thomas Osegowitsch, "The Art and Science of Synergy: The Case of the Auto Industry", *Business Horizons,* March-April, 2001.

[6] Johannes I. M. Halman, Adrian P. Hofer, and Wim van Vuuren, "Platform-Driven Development of Product Families: Linking Theory with Practice", *Journal of Product Innovation Management,* vol 20, n°2, 2003.

[7] François Fourcade and Christophe Midler, "La modularisation automobile: enjeux et conditions de mise en oeuvre pour les équipementiers de rang 1", *Gerpisa,* Onzième rencontre internationale, 2003.

[8] Mari Sako, "Modularity and Outsourcing: the nature of Co-Evolution of Product architecture and Organization architecture in the Global Automotive Industry," in ed. Andrea Prencipe, Andrew Davies, and Mike Hobday, *The Business of Systems Integration,* Oxford University Press (TBP), 2002.

[9] Benoît Weil, *Conception collective, coordination et savoirs, les rationalisations de la conception automobile,* Thèse de doctorat de l'Ecole Nationale des Mines de Paris

[10] Thomas Hellström, Ulf Malmqvist, and Jon Mikaelsson, "Decentralizing knowledge, managing knowledge work in a software engineering firm", *Journal of High Tehnology Management Research,* vol. 12, 2001.

[11] Durward Sobek, Jeffrey K. Liker, and Allen C. Ward, "Another look at how Toyota integrates product development", *Harvard Business Review*, July-August, 1998.

[12] Ron Sanchez, "Modular Architectures in the Marketing Process", *Journal of Marketing,* vol. 63, special issue, 1999.

[13] Ron Sanchez and RObert P. Collins, "Competing - and Learning - in modular markets", *Long Range Planning,* vol. 34, 2001.

[14] Ron Sanchez and Joseph T. Mahoney, "Modularity, flexibility, and knowledge management in product and organization design", *Strategic Management Journal,* vol. 17 (Winter Special issue), 1996.

[15] Franck Aggeri and Blanche Segrestin, "To what extent is it possible to combine lead times reduction and innovation ? A collective learning perspective on a recent automobile development project", in *EGOS* (Lyon: EML, 2001).

[16] Franck Aggeri and Blanche Segrestin, “Comment concilier innovation et réduction des délais ? Quelques leçons tirées du développement de la Laguna II”, *Gérer et Comprendre, Annales des Mines,* vol. 67, mars, 2002.

[17] Blanche Segrestin, Philippe Lefebvre, and Benoît Weil, “The role of design regimes in the coordination of competencies and the conditions for inter-firms cooperation”, *International Journal of Automative Technology and Management,* vol. 2, n°1, 2002.

[18] Annabelle Gawer and Michael A. Cusumano, *Platform Leadership. How Intel, Microsoft and Cisco drive industry innovation,* Harvard Business School Publishing, 2002.

[19] Rebecca M. Henderson and Kim B. Clark, “Architectural innovation : the reconfiguration of existing product technologies and the failure of established firms”, *Administrative Science Quaterly,* vol. 35, 1990.

[20] Karl Ulrich, “The Role of Product Architecture in the Manufacturing Firm”, *Research Policy,* Vol. 24, 1995.

[21] Kim B. Clark, “The interaction of design hierarchies and market concepts in technological evolution”, *Research Policy,* vol. 14, 1985.

[22] Takahiro Fujimoto, *The Evolution of a manufacturing System at Toyota,* Oxford University Press, 1999.

[23] Takahiro Fujimoto, “Thinking of the Chinese Manufactuing sector in Architecture Context”, in *REITI* (2002).

[24] William J. Abernathy and James M. Utterback, “Patterns of industrial innovation”, *Technology Review,* vol. 2, n°80, 1978.

[25] Takohiro Fujimoto and Stefan Thomke, “The effect of "Front-loading" problem solving on product development performance”, *The Journal of Product Innovation Management,* vol. 17, 2000.

[26] Gerhard Pahl and Wolfang Beitz, *Engineering Design : a systematic approach,* The Design Council, 1977.

[27] Pascal Le Masson, Armand Hatchuel, and Benoît Weil, “Le co-développement à l'épreuve de l'innovation intensive : vers de nouvelles formes d'organisation de la conception innovante entre constructeurs et équipementiers”, in *Gerpisa* (Paris: 2000).