An Analysis of the Impact of Modularization and Standardization of Vehicles Electronics Architecture on the Automotive Industry

Christophe L. Gaillard

Mechanical Systems Engineering Diploma
Université de Technologie de Compiègne

Master of Science, Mechanical Engineering
The Ohio State University

Submitted to the System Design & Management Program in Partial Fulfillment of the Requirements for the Degree of Master of Science in Engineering & Management

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

January 18, 2005

Signature of Author ....
Christophe L. Gaillard
System Design & Management Program
December 2005

Certified by ....
Daniel Whitney
Thesis Supervisor
Senior Lecturer, Engineering Systems Division

Accepted by ....
Patrick Hale
Director, SDM Fellows Program

Accepted by ....
John Grace
Industry Co-Director, SDM

To Claudine, my little angel

Acknowledgments

The pages that follow do not only represent a thesis work. They embody the learning that I have had the opportunity to do during the past two years at the System Design & Management program. The opportunity to take part of this program is what I am most thankful for. Indeed, I have had the chance to meet and work with truly interesting people and this allowed me to develop as a professional. I would therefore like to thank first and foremost my fellow SDM students, particularly those of the class of 2004. The experience of the past two years, both at a distance and on campus with them has been truly invaluable.

Patrick Hale, director of the program, and Ted Hoppe, program coordinator, have allowed this experience to be an enriching one by being quite flexible and helpful.

I am particularly grateful to ArvinMeritor, my employer, for giving me the opportunity to enter this program, and granting me the flexibility I needed in order to balance professional and school work. Jack Grace, who was vice-president of engineering at ArvinMeritor and is now industry co-director of SDM, has always been very accessible and a great resource when it comes to systems thinking.

I would like to thank Daniel Whitney, my advisor, for his insight and guidance in my thesis. Professor Whitney has influenced my work for this thesis and beyond.

I am thankful to Carliss Baldwin from Harvard Business School whose work on modularity helped shape the work that follows and whose feedback was indeed of great help.

This thesis would not have been possible without the invaluable feedback from all the people who agreed to give me some of their time. I am particularly indebted to Tom Keim from MIT, Edie Khan, Sean Newell and Timothy Thomas from the Ford Motor Company, Jelani Ellington, Mark Shaefer and Coleman Jones from General Motors, Benoit Marguet from Airbus, Steve Bresnahan from Hamilton Sundstrand, Malvern Atherton from Rolls Royce engines.

I truly appreciate the help that Fernando Cela Diaz, Evangelos Mamas and William Taylor, fellow SDM students and alumni, provided for this thesis, but also at various other occasions.

I would like to especially thank my wife, Souri, for her support and patience with me for the past two years.

Finally and most importantly, through difficulties, my daughter Claudine, by her incredible hunger for life and despite the challenges she has been facing early on, shows me what is most important.

Abstract

The growing use of electronics in automobiles designs and their dependency on it, has increased the level of complexity of the car-system and created new challenges. But at the same time, it has created new opportunities and the potential to reduce complexity through modularization. This represents a new architectural paradigm for OEMs and suppliers.

This thesis suggests an approach to this new era of automobiles designs. It looks at the effect of modularization and the advent of electronics on the supply chain in other industries. It evaluates the risks of value migration in the automotive industry and studies the mechanisms of such migrations through several interviews, financial data research, systems functional decomposition and system dynamics analysis.

Electronics, along with software and control algorithms, enable an encapsulation of functionalities by creating higher levels of abstraction. While early vehicles had an all-mechanical interface between operator and actuation, electronics has allowed the separation of the processing of signals coming from the operator, the control/functionality infusion, the transfer of information and the transfer of energy. Thus, what was once integral has now the potential to be modular. Such a separation increases the modularizability of the automobile's architecture and gives it an opportunity to get closer to a lower bound "essential complexity" floor.

While integrality helps prevent knowledge from fleeing away, it limits the ability to profit from various design options. When outsourcing for a modular architecture, those exclusive functionalities that actually bring value ought to be retained in-house. In particular, outsourcing software is usually not a desirable option. Software modules are likely to remain intrinsically integral for a long time to come. OEMs should thus look at expanding their software expertise in order to eliminate any dependency to an outside source for software, because it is likely a dependency on knowledge.

Suppliers, who have already taken on a greater system integration responsibility, should look outside the traditional mechanical systems box. Automotive systems today

involve electric, electronics and software engineering. To gain the necessary expertise in those domains, suppliers may have to perform strategic mergers & acquisitions.

The role of system engineering is what OEMs ought to focus on if they want to avoid seeing value migrate to their suppliers. The emergence of value is the fruit of architecting. New open standards should be regarded as opportunities to become more aggressive systems architects. Open standards also allow to reduce cost, in particular by creating economies of scale and scope. However, reducing cost without creating value is the beginning of a downward spiral.

Modularization and standardization have created a dynamic reaction in the industry whereby the nature of the boundaries between firms is changing and value is created and redistributed. In order to capture that value, a player has to focus on the design process, the architecting of products, rather than on the products themselves. The role of system architect in the automotive industry has evolved and now requires expertise in the field of software development, testing and integration.

Table of Contents

Table of Figures

I. Introduction

Since the early ages of the automotive industry, automobiles have been mainly mechanical machines and have evolved as such. However, this paradigm is currently shifting and many innovations in cars are now oriented towards electronics. In 2004, 19% of a vehicle's total cost was accounted for by electronics, but this number is expected to rise even further – some analysts predict it to be 40% by 2015.

While electronics appears to be more and more pervasive, it is not quite clear how this will affect the overall architecture of cars. Such innovations as brake-by-wire are in their infancy but may prove to have radical influences on what the car of tomorrow will look like. By impacting the architecture of automobiles, electronics is likely to also impact the supply chain architecture. As vehicles architectures become more modular, so too may the supply chain architecture become more horizontally integrated.

One objective of the proposed work is to study how electronics may affect how cars are architected. Perhaps, what electronics can enable and how the industry players may make use of it are two different things. Indeed, certain automotive companies seem to introduce electronic innovations into their cars for the simple sake of adding technology. But those innovations do not always respond to customers' needs, their integration is not optimal, and the effect on the brand is in fact negative. The total number of recalls in Germany has grown on average 8% from 1995 to 2003. In comparison, the number of electronics related recalls has grown on average 17%. Clearly, there is room for improvement and a better architecture may provide a good background for better integration of electronics in vehicles.

By taking a holistic view of the whole system architecture, this work proposes to study how the supply chain may be affected by such an evolution. The aircraft industry has already shifted to fly-by-wire technology and single databus communication architecture and has taken advantage of the flexibility in design that this has brought; the PC industry has experienced a complete overhaul as a result of IBM modularizing its architecture and opening it to its suppliers, as a result, the industry has evolved from a vertically integrated model to a more horizontal one.

With more and more computer technology inside a car, one could wonder for example if a car could potentially become a computer on wheels; could it ever be similar in modularity to a PC where components may be sourced independently, yet still be integrated?

Finally, this thesis will look through the “disruptive technology” lens to examine to what extent the pervasiveness of electronics in automobiles may affect the automotive industry. What model of disruption may it follow? What are the patterns observed that may give us a hint of what is to be expected? What or who is being disrupted? And where do the threats come from?

II. Automotive Electronics

II.A. Current State

It is widely recognized that electronics content in passenger vehicles is growing increasingly. According to reports, the total World market for automotive electronics is forecasted to grow from $23.0 billion in 2003 to $42.5 billion in 2012 (CAGR of 7.1%).[1] Even though the automotive market may suffer from a lower growth, the demand from OEM is expected to increase faster than customers' demand for cars, which indicates that the average electronics content in vehicles is growing. In fact, it is estimated that 80% of all innovations between now and 2015 in the automotive industry will be electronics based. As a result, the average passenger car that contained $410 worth of electronic control units (ECU) in 1995 contains $680 today, and will reach about $860 by 2015[2].

An average car contains approximately 5000 passive electronic components, while luxury vehicles may have as much as twice that number. While powertrain related controls – such as engine, transmission, fuel injection – have made use of electronics for some time (the first engine control chip went into production in 1980), the growing demand for more sophisticated safety, entertainment and communication functionalities is driving the increase in the market outlook for automotive electronics. Hybrid powertrains also add to the novelty of electronic functions.

Figure 1 lists the results of a survey by J.D. Power & associates aiming at determining what the most wanted new technologies in passenger vehicles are. Although the feature ranked number 1 is not related to electronics, it is interesting to notice that most of the others are.

The 4 top ranked technologies are safety related. After that, it is a mix between convenience, infotainment and safety.

In particular, the following functions are expected to grow more than 40% within the next 5 years: adaptive front lighting; starter-alternators; passive entry systems; tirepressure monitoring systems; electric parking brakes; head-up displays; electric hybrid drivetrains; active front steering; and electronically controlled manual transmissions.[2]

The prices mentioned on this list point out an interesting fact: the cheapest features are those where electronic piggy-backs already existing components and comes as an enabler of new functionalities. For example, windows obviously exist in every car, and most cars today have electrically operated window regulators, so the smart sensing window feature only needs to add a relatively simple sensor and a control component to activate the window regulator. In contrast, night vision requires a more elaborate detection system, a sophisticated display system, etc. hence its cost is one of the highest.

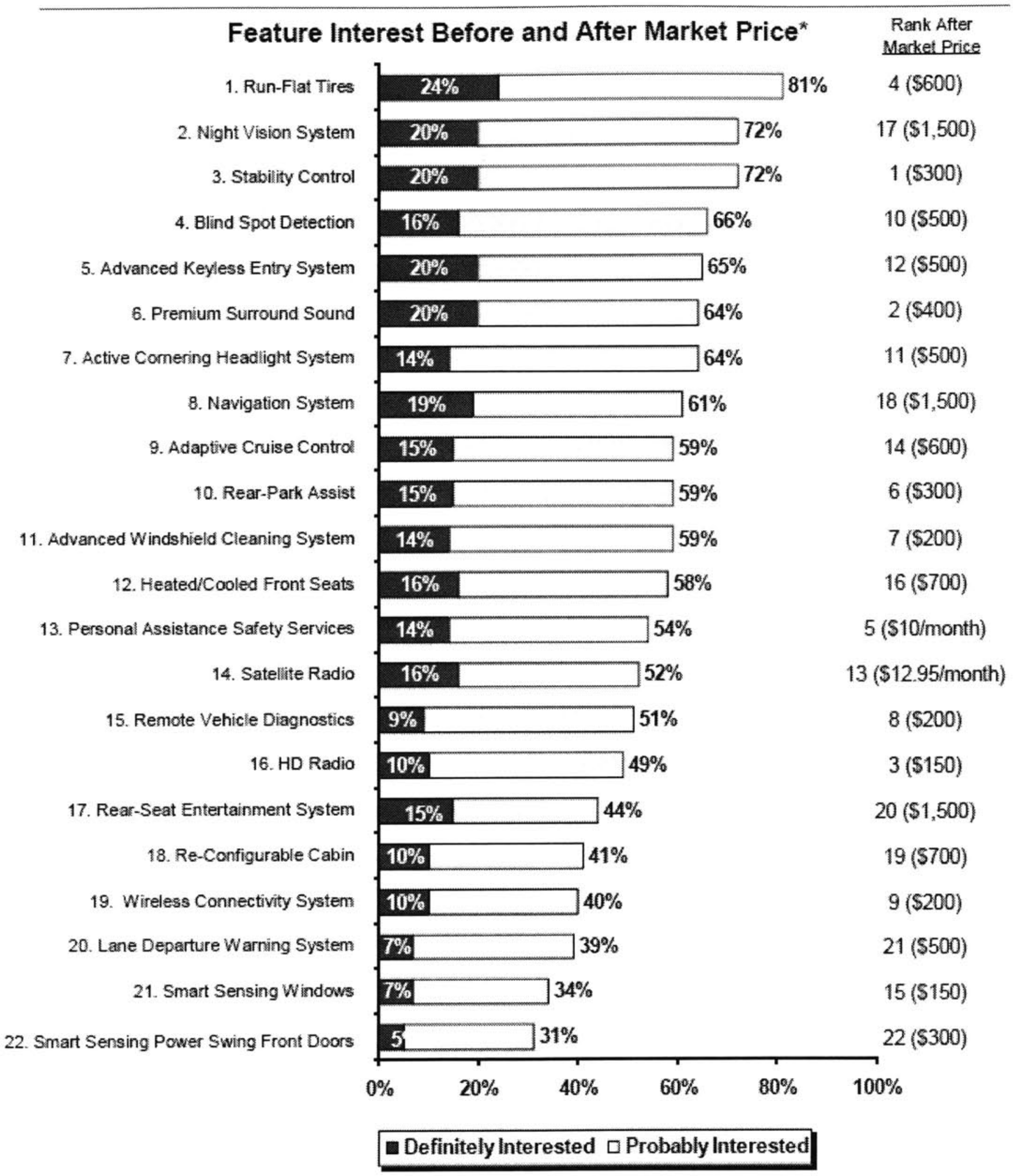


Figure 1. Emerging technologies consumers are most interested in (from J.D. Power & Associates[3])

II.B. Electronics & Complexity

"...the auto industry, producer of the most complex mechanism that human intelligence has ever devised for mass consumption"[4]

- Fred Andrews

Frenchman Nicolas-Joseph Cugnot is credited for engineering the first self-propelled vehicle in 1769. It was a three-wheeled "wagon" with a steam engine installed on it (Figure 2). In the two and a half centuries that separate us from Cugnot's first vehicle, the design of automobiles has evolved in many ways. Steam engines led to internal combustion engines in 1886 (independently developed by Carl Benz, Gottlieb Daimler, Wilhelm Maybach and Siegfried Marcus). With the advent of electricity, electric motors became popular in the early 1900s because they were easier to start (no hand crank), easier to drive without gears to shift, it was quieter and produced less vibrations and less smell. But with the development of road networks, the limitations of electric motors in terms of range and speed became obvious. Gasoline became the dominant design and many improvements were subsequently made – multi-valve, overhead camshafts. With increased speed came innovations on braking systems with the disc brakes. Driveshaft eventually replaced drive chains. Electric ignition and electric self-starter appeared to make it easier to start the car, suspensions were improved for enhanced comfort, transmissions and throttle control allowed for a wider range of speeds.

With increasing vehicle speed requiring enhancements for safety and comfort reasons, demand from customers for more comfort, increased competition thanks to the ease for foreign makers to enter new markets, and eventually the need for better fuel efficiency, manufacturers have developed new technologies in abundance in the automotive industries, often leading to new constraints and more interactions between various components and systems. As a result, cars have become very complex systems.

Furthermore, in the past ten to fifteen years, electronics has made its way into automobiles designs. While this allowed enhanced control of various systems, it also added new dimensions in the complexity of the overall architecture of automobiles as it permits to link various systems together.

The car as we know it today is indeed quite complex, one simply has to open the hood of a vehicle or look at its underbody and see the myriad of cables, hydraulic lines and linkages to realize that today's car is far more complex than Cugnot's wagon and steam engine machine.

In the following section, we discuss what complexity really entails as we will later see that electronics and complexity go hand in hand.

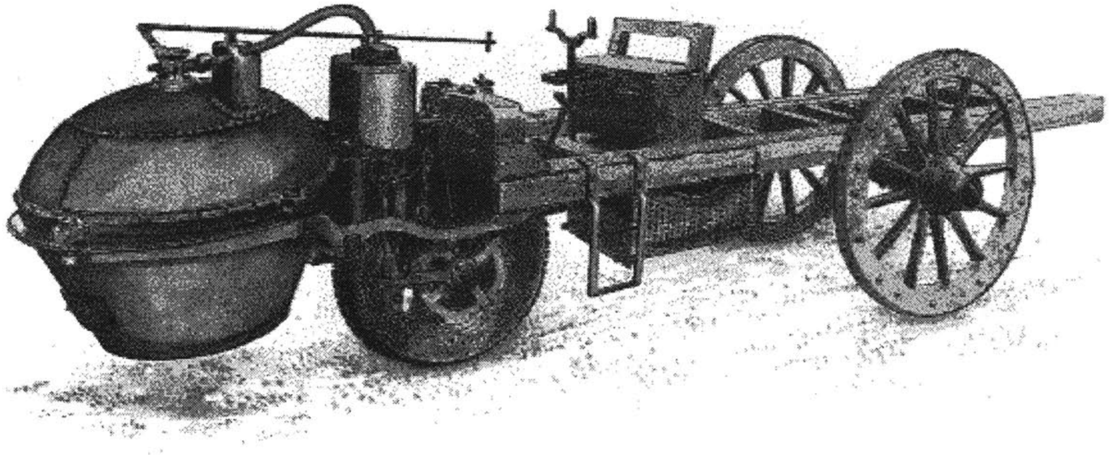


Figure 2. Cugnot's automobile - a three wheeled "vehicle" powered by a steam engine (from *Inventors*[5])

II.B.1. Complexity vs. Complicatedness

Complex and complicated are two different adjectives that have distinct meanings. It is indeed possible for a complex system to be uncomplicated and for a not so complex system to be complicated.

In fact, the automobile is a good example to illustrate this difference. The automobile is a complex system/product, but it is not complicated – for the driver that is. Only three pedals, a gear shifter and a steering wheel suffice to control the vehicle. And it is even simpler with automatic transmission.

It may however become complicated when looking at it from the perspective of the mechanics who wants to repair it, or the manufacturing group, etc. Few drivers are aware of how complex a car really is. Even when broken down in areas such as chassis, powertrain, electronics/electric, body, interior, exterior, etc. there is clearly still a lot of complexity within each area, and much interaction between each one of them.

There are various dimensions of complexity. Durability analysis, acoustics & vibration, safety related work may all create conflicting objectives for various components or sub-systems that interact with each other; this represents a complexity at the requirement or performance level. Another dimension relates to how components and/or sub-systems interact with each other to perform often multiple functions; this is the functional complexity. Yet another dimension is the form complexity that comes from having many parts that need to somehow communicate and interact together; this is a topological complexity. Because of the huge number of components, requirements and functions in a vehicle, the level of complexity is overwhelming.

This brings forth an essential point: how complex and/or complicated a product is depends entirely from what point of view one looks at it. Complexity is usually increased by the number of subsystems that make-up and interact within the entire product and how many subsystems layers exist. Complicatedness however is a function of the interrelationship between these subsystems. Therefore, if boundaries between subsystems are well managed, a complex system does not necessarily turn into a complicated one. This is why a car is not complicated for a driver, but it is rather complex and potentially complicated for a manufacturer and its suppliers who have to make all the pieces fit together.

According to Professor Edward Crawley, "Interface definition and control are an area of prime importance to the system architect" because "interfaces

- define the "system nature" of the product
- are an area of great leverage
- are an area of great uncertainty
- are key to
  - assembly/integration/checkout
  - maintainability
  - product evolution and adaptability"[6]

In the automobile's case, the boundary between the driver and the car's commands are relatively well standardized: acceleration is controlled by the right foot via a pedal on the right hand side, clutch activation is activated by the left foot via another pedal on the left hand side, braking is controlled by the right foot via another pedal in the middle, steering is controlled with a wheel positioned at chest level in front of the driver and turn signal is controlled by a lever on the left hand side of the steering wheel. Because these commands are standardized, a driver can switch from one car to another and easily and rapidly adapt, but also, because controls are always the same, a driver tends to perform the necessary actions to drive a car without even thinking about it.

Hence, we see that complexity and complicatedness reside at the boundaries. The complicatedness of driving the car is significantly reduced by using the same standard interfaces to the driver.

But this is the external layer of boundaries only and it affects only the driver-customer. It is because the interdependencies at that layer are kept to a minimum and are well understood that driving a car is a simple uncomplicated process.

When we dive into more internal layers of boundaries however, interdependencies may become much more complex and involve many subsystems at the same time.

"You have the potential with the controller to add additional functionality. Today for example, the controller interfaces with the engine control module to slow the engine down, and it has the ability to interface as well with the steering system and the suspension system for added functionality" said William L. Kozyra, President and CEO, Continental Automotive Systems North America[7]. This is just an example of the many interactions that exist between various subsystems.

When Henry Ford started building cars in the early 20th century, breaking down the vehicles in chassis, cabin and powertrain may have provided a way to organize the production and development and in turn the enterprise structure as a whole in a relatively efficient fashion at the time. In effect, the architecture of the vehicle was quite modular in the sense that each element could be developed relatively independently and assembled together late in the process (this is a modularity in production, see III.A.2). But in the modern era of car development, new requirements, new technologies and new performance targets have created much more interactions between sub-systems. For example, acoustics and vibration or durability performance targets require one to think of the physical interactions between components and how and where the energy is transferred. The engine, for example, may be a source of vibration, but engine mounts are critical to how those vibrations are transmitted to the chassis; boundaries between the manifold and the exhaust pipe is another one. And as William Kozyra's example shows, the advent of automotive electronics gives the opportunity to bring different components together in providing a function.

In fact, Dan Whitney argues that there is a physical limit to modularity[8]. He explains that this limit is linked to the amount of power that is required to operate the product, the key issue being the back-loading that such power engenders, thus forcing one to consider all the interactions between every component. The problem is that these interactions are not controlled by design but by laws of nature, they are often unwanted, and many of the interfaces existing between components or modules are "unidentified". The result is unwanted emergent behavior and the product has to be controlled and tested at the system level. Controlling each module independently is not enough to satisfy that the system will work properly because it is almost impossible to predict perfectly how each module will interact with other modules.

He argues therefore that (Very Large Scale Integration) VLSI systems such as computer chips can be designed, used and produced modularly because the interaction between each component and between each of them is purely logical, informational, and the amount of energy is almost null. It is thus possible to mathematically predict the behavior of each part and of the whole system.

One may therefore say that such systems can be quite complex - because there are indeed many interactions between each component, but it is not complicated - because these interactions can be modeled, predicted without ambiguity and no emergent behavior is to be expected. We come back to this point in §IV

II.B.2. Systems vs. Modules

Although the terms “system” and “module” are often used interchangeably, they have fundamentally different meaning and it is important to define them here.

A module is essentially a set of components that are physically assembled together and delivered as a whole so that the customer can simply take it out of a box and install it in or on its product.

In contrast, a system is a set of components that, together, are designed to deliver a certain functionality.

In other words, the bounds that exist between components in a module are in the physical domain, and the bounds between components in a system are in the functional domain.

A system may be distributed over various modules, and a module may contain elements from various systems.

The understanding of the difference between the two terms also helps clarify how complicatedness arises. Indeed, when an architecture is modularized to the point where a module is a self-contained system and a system is physically concentrated into a module, then the level of complexity is much less than if the architecture were more integral and systems were distributed over various modules and modules contained components (or subsystems) from various systems.

II.B.3. Managing Complexity

As we have seen in the previous section, the number of interfaces is a key determinant of how complex a system is and the nature of these interfaces is a determinant of how complicated the system is. Because of this, although there may be an intrinsic complexity to a particular system, it is more useful to try and understand what the level of complexity of a system is for a given protagonist, be it the driver/customer, the manufacturer or the suppliers.

Professors Charles Fine and Dan Whitney[9] explain that the ultimate core competency within a company is to know what competency to retain in-house and what competency to outsource. And in order to do so, the company must be able to organize its products in modules or elements in a way that the interface between each of them minimizes the parameters that link them to each other and make them clear and unambiguous. This in turn translates into how suppliers (internal or external) can effectively work with one another. Complex interactions are to be kept within each module where there is a tight cooperation among the development team.

In fact, Fine and Whitney argue that the competency for product development process, systems engineering, product architecture and modularity, and supply chain design are the same. According to them, systems engineering is the top-down process of engineering an integral system. As the system becomes more and more modular (i.e. the functions are more and more independent of each other), the need for systems engineering becomes less and less. That is because the interfaces have been standardized, and therefore, there is a universal understanding of how one module communicates with another.

One may thus see a standard modularization of a system as "pre-packaged" system engineering. There is a common agreement on how the system is to be broken down, on what the high-level architecture and how interfaces are defined.

Regardless of the level at which complexity has to be managed – industry, company or division – the most fundamental and essential skill required is the same: system decomposability.

Fine & Whitney explain that when a company makes the decision to outsource a component, module or system to a supplier, it accepts to be dependent on that supplier. They provide a truly invaluable framework for studying the make-buy decision process by determining the following two types of dependencies:

- dependency on capacity and
- dependency on knowledge

In the former, the company decides to outsource because it may be more economical to do so, but it has the knowledge to make the part or system. In the latter case, however, the company could not make it even if it had the money, time and capacity to do so.

The importance of this approach lies in the fact that outsourced components have to be assembled together to form a system. If the system is truly and purely modular, those components may be a black box. As long as it is understood how the components interface with each other, there is no systems integration, and the task is fairly simple. However, for an integral system, the system engineering task is essential, but cannot be carried with black boxes. This is because the "unidentified interfaces" will remain unidentified.

Therefore, one can see that dependency on knowledge is acceptable if the interaction between components is simple and clearly understood. However, dependency on knowledge is not acceptable for components where interactions are multiple and it is not possible to precisely predict how they will interface with each other.

In the automotive industry, in order to manage the increasing complexity of car design, manufacturing and assembly, OEMs have given their suppliers more responsibility over the past decade and asked them to supply modules and systems rather than just components. The decomposability competency is therefore crucial and the framework offered by Whitney and Fine is very useful to think about how to go about this decomposition. However, it is not clear how deeply into the decomposition layers one should go. In other words, if a system is decomposable into a maximum of n modules, should it be decomposed into n modules, or should it be decomposed in fewer bigger modules which in turn may or may not be decomposed into sub-modules depending on the supplier?

The stance that many automotive OEMs have adopted is to decompose the system into fewer, bigger modules. The idea is that with fewer modules, there are fewer interactions to manage, hence reduced complexity. The problem is that with increased outsourcing of modules and systems, if a minimum of knowledge is not kept in-house, then each module represents a black-box that is difficult to integrate into the system. Bigger modules mean that the customer only controls the higher level interface. Other interfaces are hidden and under the control of the supplier. As professor Whitney explains, "this works only if the system really is modular, that is, that the only interfaces are the ones that were designed in, and that there are no others"[10]. He also points out that such a case almost never exists. Indeed, particularly with a system involving high energy transfers, undesired behaviors at the boundaries emerge. As a result, it is a dangerous thing to do that to leave the control of lower levels interfaces to suppliers.

Toyota for example, known for its excellence in automotive manufacturing and assembly, has so far outsourced virtually no module or system to its keiretsu suppliers, preferring to assemble its own modules in-house. It is only now evolving toward more module outsourcing because it is observing that competitors such as Nissan and Honda are leveraging this practice for a cost advantage.

There is therefore a compromise to be found between too much decomposition whereby control of the interfaces and thus of the system is lost, and not enough whereby economic advantages are not leveraged.

According to Professor Nam Suh, an ideal design is uncoupled or decoupled in that it maps one functional requirement to one design variable[11]. Although it is in theory not linked to the notion of modularity vs. integrality of a design, it does promote a modular decomposition. Armen Zakarian suggests a cost-based approach to deciding how and how much a system should be broken down into modules[12]. Baldwin & Clark balance this approach by considering modules as value options and demonstrating that more modules means more value potential on one hand, and also the fact that more modularization often means higher cost as well, particularly cost of experimentation.

In summary, it is apparent that many factors ought to come into the equation of the modularization of a system or architecture:

- Strategy
    - When does outsourcing mean that knowledge will vanish?
    - When is it ok to share or potentially lose knowledge?
- Cost
    - What investment is required in order to change the architecture?
    - How much experimentation is necessary to validate it?
    - If modular architecture means standard interface, what cost savings are available through standard module outsourcing?
- Value
    - How does the modularization of the system make the system more customizable? More attractive? More valuable for the customer, for the OEM, for the supplier?
    - How do improved component designs actually improve the system?

II.B.4. Automotive Electronics

II.B.4.a) The Dilemma of Automotive Electronics

Pervasiness of Electronics in Cars

According to McKinsey, mechatronic components (i.e. those sub-systems that include electronic, electric and mechanical parts) accounted for less than 1% of the total cost of a vehicle in 1980. That number has risen to 20% today and is expected to reach 40% by 2015[13].

The aircraft industry has known a similar fate with the transformation of planes' command architecture from hydraulic to electric, also known as fly-by-wire. The first planes to be designed with fly-by-wire were military aircrafts in the 1940s and 1950s. The original goal of designing an autopilot function led to the realization that all flight controls could be operated via an electric rather than hydraulic network. The advantages were multiple; a significant gain in weight due to the disappearance of bulky hydraulic lines, pumps, etc; less mechanical parts that could potentially wear, break and fail.

Soon, designers realized that the use of an electrical/electronic (E/E) network to link sensors, actuators, control units and computers provided many opportunities. In particular, it gave the possibility to reduce the impact of human error by creating boundaries around an operating envelope that could be precisely tailored and that the pilots could therefore not cross – Boeing and Airbus actually have different philosophies about the use of fly-by-wire with respect to human control overriding. For example, computers on Airbus planes prevent the pilot to put the plane into a climb of more than 30° which may cause the plane to lose lift and stall, maximum nose-down pitch is 15°, maximum bank or roll allowed is 67°. Any maneuver that would create acceleration in excess of 2.5g is also prevented. Boeing took a different approach with regards to fly-by-wire about who has ultimate control over the flight commands. Indeed, while in an Airbus plane, a pilot is constrained by the flight envelope and computers retain ultimate control, in a Boeing plane, the pilot can override this envelope if he/she deems it necessary. These are two fundamental differences in the philosophy that may be adopted with regards to what roles computer ought to play for safety critical functions.

Fly-by-wire also allowed fighter jets to be designed with very low natural stability (relaxed stability). Low natural stability means that the plane can roll, yaw and pitch very quickly and with much less effort, however, it obviously also means that it is much more difficult for the pilot to control the plane. The fly-by-wire architecture allowed the design of automatic controls that are seamless to the pilot. The result is drastically improved maneuverability. Electronics allowed to get the best of both worlds: control and maneuverability.

Moreover, customization of the aircraft's behavior was possible. In particular, controls allowed increased smoothness in the response of the plane to pilot inputs, oscillation avoidance, etc. While this architecture was first developed for fighter jets, it made it into commercial airplanes in 1976 with the Concorde – a supersonic airliner developed jointly between Aérospatiale and British Aircraft Corporation; then with the Airbus A320 in 1988 and subsequent Airbus planes. Boeing adopted the fly-by-wire architecture with its 777 a few years later.

This historical overview of the adoption of the fly-by-wire architecture in aircraft designs shows the benefits that increased electronics brought to that industry.

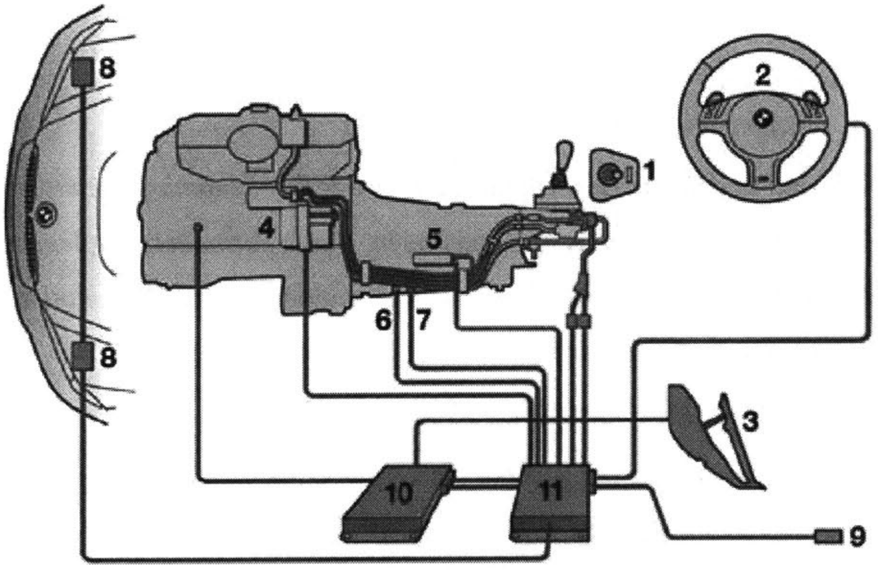
Electronic components in automobiles are not new. Several functionalities already exist that depend on embedded electronic parts. Below are a few examples of functionalities driven by the advent of electronics technologies.

1. <u>BMW's Sequential M Gearshift (SMG)</u> is a system that uses a dry friction clutch like those used in manual transmissions, but it automates the gear shift from one touch of a paddle (right for up-shift and left for down-shift). When the paddle is pressed, an actuator disengages the clutch plates, then another one shifts gears, the engine speeds is adjusted to ensure that it matches the exact rotation of the output plates given the vehicle speed and the gear ratio, then the clutch is re-engaged. There are many advantages to this system: there are less moving mechanical and hydraulic parts which means less costs and wear and less potential failures from wear. The complete gear shift from the moment the paddle is pressed takes about 80 milliseconds, faster than most human driver can accomplish and without the

possibility to miss a gear. The gear shift feel can be customized for smoother or sportier styles or even winter/slippery surface conditions. It can be switched to complete automatic mode where an algorithm decides when to shift, just like a conventional hydraulic converter automatic transmission. Note on Figure 3 the electronic linkages that exist between various control modules, sensors and actuators.

2. Active pedestrian protection is yet another safety feature that OEMs are currently working on developing. It is particularly geared toward vehicles with bigger engines because spatial constraints. Indeed a smaller engine means more space under the hood; this allows the hood to be deformed thus expanding the design space for the development of energy absorbing materials. With bigger engines however, where there is little or no space under the hood, solutions are proposed, including a design with actuators that would raise the hood as a pedestrian is about to hit it. This requires sensors to detect the pedestrian and complex algorithms to distinguish between pedestrians, other objects and vehicles.[13]
3. Electronic Stability Control (ESC) like Anti-lock Braking System (ABS) is a safety system that reads speed sensors at each wheel, compare those speeds to each other as well as to other inputs such as the actual speed of the vehicle, and (specifically for ESC) steering angle from a sensor that measures the rotation of the vehicle around the vertical axis. In both cases, those sensor outputs are used into algorithms that control the pressure of the brake at each wheel thereby correcting the behavior of the vehicle according to certain characteristics of what the optimal behavior ought to be.[14]
4. Rollover prevention is a functionality enabled by adding the signal generated by a gyro sensor that detects the rotation of the vehicle around its longitudinal axis into the ABS algorithm. Brake pressure can therefore be applied appropriately when the algorithm detects that the vehicle is approaching the rollover limit.
5. Illustrating the convergence of all technologies and domains through electronic, Nissan is developing a system that incorporates global positional and traffic information into optimization algorithms for hybrid powertrain energy use. An

algorithm uses the position of the vehicle with respect to the topology of the roads it is driving on to extract road grade information and uses live traffic information as well to predict how the power of electric batteries is going to be used (is it likely to be recharged through regen or discharge because of the need for acceleration?). By feeding back this information to the battery state of charge (SOC), it enables a better, predictive management of the battery charge/discharge which translates into better fuel efficiency.[15]



1. Drivelogic control
2. Paddles
3. Accelerator input
4. Hydraulic unit
5. Clutch-position sensor
6. Input-shaft-speed sensor
7. Transmission-oil-temperature sensor
8. Hood switches
9. Longitudinal-acceleration sensor
10. Engine-control module
11. SMG-control module

Figure 3. BMW SMG II system overview (from Motortrend.com [16])

The list of electronics enabled technologies is quite long, but these previous examples show that there are several categories of benefits including:

- Active safety (accident prevention)
- Passive safety (protection in case of accident)

- Cost savings (increased fuel efficiency, reduction of number of parts)
- Decrease of mechanical wear
- Weight savings
- Customer experience enhancement

On the other hand, the complexity generated by such systems is obvious. In particular, Figure 3 depicts the interdependencies that exist between the components but also between various domains: mechanical, "frictionless", controls…

Software: Where the Power Resides

As Paul Hansen wrote "*Functional software is fast becoming the most essential ingredient in automotive electronics, the means by which carmakers implement distinguishing features, the features that add to the vehicle's value in the mind of consumers. Functional software helps a vehicle stop faster and steer better. It improves fuel economy and makes a vehicle safer and more fun to drive*"[17].

While software accounts for about 2% of a vehicle manufacturing cost today, that number is expected to grow to about 7% by 2015[18].

Electronic control units (ECU) have been the key constituent within the automotive electronic components. This is where functionalities reside. An example of a typical ECU is the one that controls fuel injection and ignition systems of an engine. In order to do so, the ECU reads the outputs from several sensors and uses them in an algorithm.

As software programs become more external and independent of infrastructure, the role of ECUs becomes more common. They will be microprocessors supporting software program applications as supposed to a unit with embedded software. With such a shift, the creation of functionalities is much more flexible, allowing better customization, similar to what occurred in the aircraft industry. Moreover, with control being turned over to the software domain, the number of ECUs may be reduced and optimized. Instead of having one ECU per function, ECUs will be connected to several sensors, and the functions will be embedded in software programs. As a result, the standardization of ECUs is likely to translate into price dropping.

All in all, this indicates that the functional value creation in automobiles seems to have shifted from physical controls (mechanical, hydraulic or even electronic) to software programs.

This is quite an important paradigm shift, because whoever owns the software is therefore quite powerful. And because software program is so intangible, it is rather difficult to reverse-engineer it, unlike mechanical or hydraulic systems.

OEMs are starting to realize how important such a shift has become. Francisco Javier Garcia Sanz who sits on the board of management of Volkswagen AG said in an interview: "*we realized that we cannot continue to buy black boxes, because we are continually getting more black boxes, and they all have to communicate much more than they did in the past. [...]*

This is one of the big fields that suppliers are very hesitant to talk about, because this is the only field that is non-material, and everything that is non-material you cannot measure. We are now starting to understand what software is"[19]

While it is obvious that, with the prevalence of software program in vehicles, suppliers who own the software have acquired a lot of power over OEMs, Mr. Sanz's comment points out that OEMs still have the responsibility to integrate all the functions coherently in the vehicle.

Beyond any migration of value creation and/or capturing, the integration of all the functionalities is the true challenge that hangs over the whole automotive industry.

The Downside of the Proliferation of Electronics

While mechatronic sub-systems represent an increasing portion of today's automobiles and an even bigger one of tomorrow's, most OEMs are increasingly relying on their suppliers to provide sub-systems or modules, but more importantly, many rely on them to perform the integration task. Figure 4 not only gives an idea of the increasing amount of electronic components, it also shows the various networks that connect these components to each other. Below is a description of the various networks that have been developed or may be used for automobile applications.

CAN: the Controller Area Network (CAN) was developed by Bosch in the mid 80s, it has been since the most widely used network. Often, there are several CAN networks in a single vehicle. One low speed CAN is dedicated to non-safety critical functions such as window control and other passenger comfort related command transmissions. A higher-speed CAN is dedicated to more time-critical applications such as engine management, skid control and ABS.

Bluetooth: It is an open standard for short-range low-power radio network. It allows simple devices such as cell phones, multimedia players, PDA, etc. to communicate with one another.

Byteflight: A flexible high-performance protocol designed for safety related applications, Byteflight can also be used for convenience functions, as well as X-by-wire applications. Its development was a collaborative effort between BMW, ELMOS, Infineon, Motorola, and Tyco EC.

D2B: The Domestic Data Bus (D2B) was developed by Philips and Matsuhita for multimedia applications.

FlexRay: A high data rate protocol proposed by BMW, DaimlerChrysler, Philips, and Motorola. It is expected to be utilized for chassis control, powertrain systems, X-by-wire applications.

IDB-C: The Automotive Multimedia Interface Collaboration, an organization that includes several OEMs, developed the Intelligent Transportation System Data Bus to enable plug-and-play of multimedia applications.

LIN: The Local Interconnect Network (LIN) is an inexpensive open standard developed by Audi, BMW, DaimlerChrysler, Motorola, Volcano and Volkswagen. It is a master-slave time-triggered protocol. It is used as a sub-network that connects such components as seats and mirrors to a CAN.

MML: The Mobile media link was designed by Delphi for automotive multimedia applications such as game consoles, navigation system, etc.

MOST: The Media Oriented System Transport (MOST) is a widely used fiber optic network protocol for multimedia networking. It was jointly founded by BMW, DaimlerChrysler, Harman/Becker and OASIS SiliconSystems, but many firms eventually joined the cooperation.

TTCAN: Time-Triggered CAN is an extension of the CAN protocol intended for chassis control, transmission, engine management and X-by-wire applications.

TTP: The Time Triggered Protocol (TTP) was designed with time- and safety-critical systems in mind. Time triggered – as supposed to event triggered – means that the connection between systems occurs systematically at specific time intervals rather than being triggered by an event or a logical statement. Therefore, if there is an error during a certain communication, there is virtually no delay since another communication event occurs very shortly afterward.

Note from this list that there are competing networks such as Byteflight, Flexray or TTCAN who could all potentially be used for x-by-wire applications, or MML, MOST or IDB-C who could all be used for multimedia applications. Nobody really knows how the adoption of these standards will evolve.

On the other hand, many OEMs (such as BMW, DaimlerChrysler) as well as electronics suppliers (such as Motorola) are involved in the development of these competing networks. It looks as though they are hedging their bet by making sure that they keep a foot in each one in case one of them emerges as a dominant standard. As soon as this happens, they want to be ready to jump on the train. This is an indication of how important this segment of the automotive business is.

But as Figure 4 shows, at least for the moment, several different networks exist in a vehicle. More importantly, there does not appear to be a system-wide standard that allows one component at the end of a network to seamlessly communicate with another component at the end of another network or even on the same network. In other words, if a supplier wants its module to communicate with a module from a different supplier, both suppliers need to get together to develop their product in concert so that they can achieve the functionality that they want. The networks are successful in transferring messages, but they do not assure that the message from one control unit will be understood by another.

The various networks provide standard infrastructures and protocols, but in most cases, software codes are specific to the supplier and to the control unit.

This clearly puts an OEM in a situation where the integration of the whole system is rather difficult. Indeed, as we have seen earlier, software programs are black boxes for them, yet everything has to be seamless for the end user.

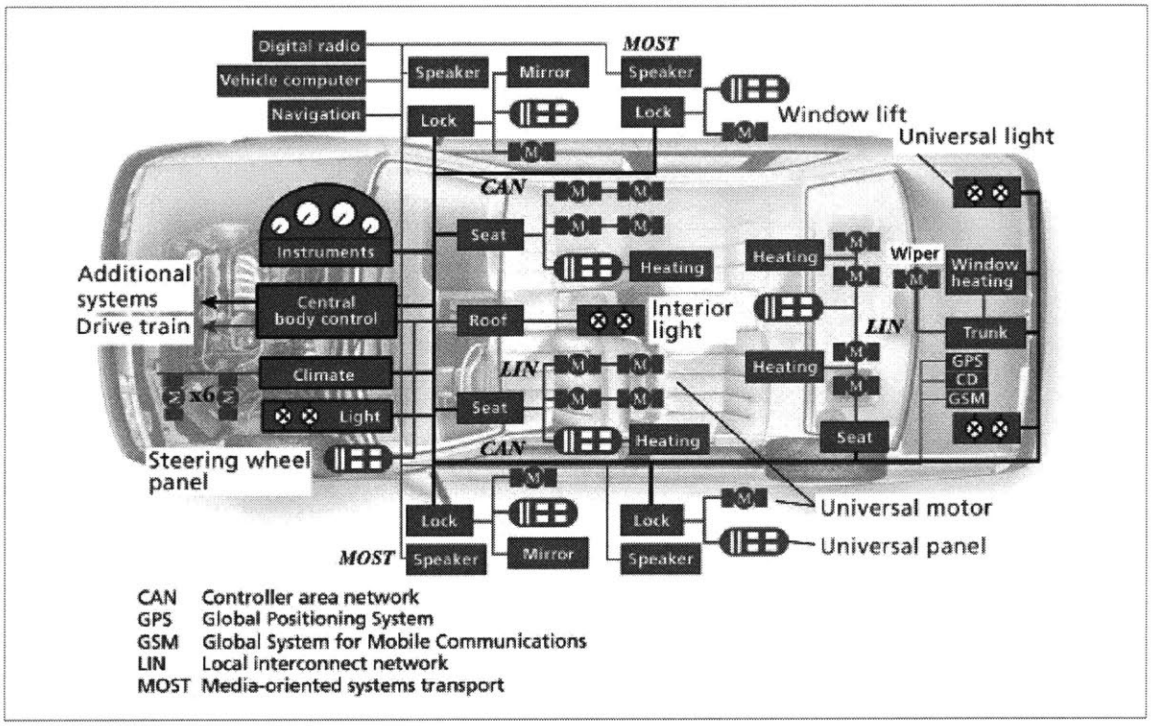


Figure 4. Representation of a passenger car network architecture (from Leen & Heffernan[20])

Quality Issues

With the various networks coexisting within a car, and most importantly the different pieces of controls, electronics and software in a vehicle, the reality seems to be that most OEMs cannot effectively manage to integrate all of it. According to the German Automobile club ADAC, 52% of all vehicle breakdowns are due to software and errors and electronics problems and that number is forecasted to get even bigger.

Figure 5 shows the history of recalls in Germany from 1995 to 2003. While non electronics related recalls have been fluctuating, their overall evolution has been somewhat stagnant as the linear regression trendline shows. The 2003 number is only 38% higher than the 1995 number which represents a compounded annual growth of 4.1%. In contrast, electronics related recalls have steadily increased over those years and the 2003 number is 250% higher than the 1995 number, which is a compounded annual growth of 16.9%.[13]

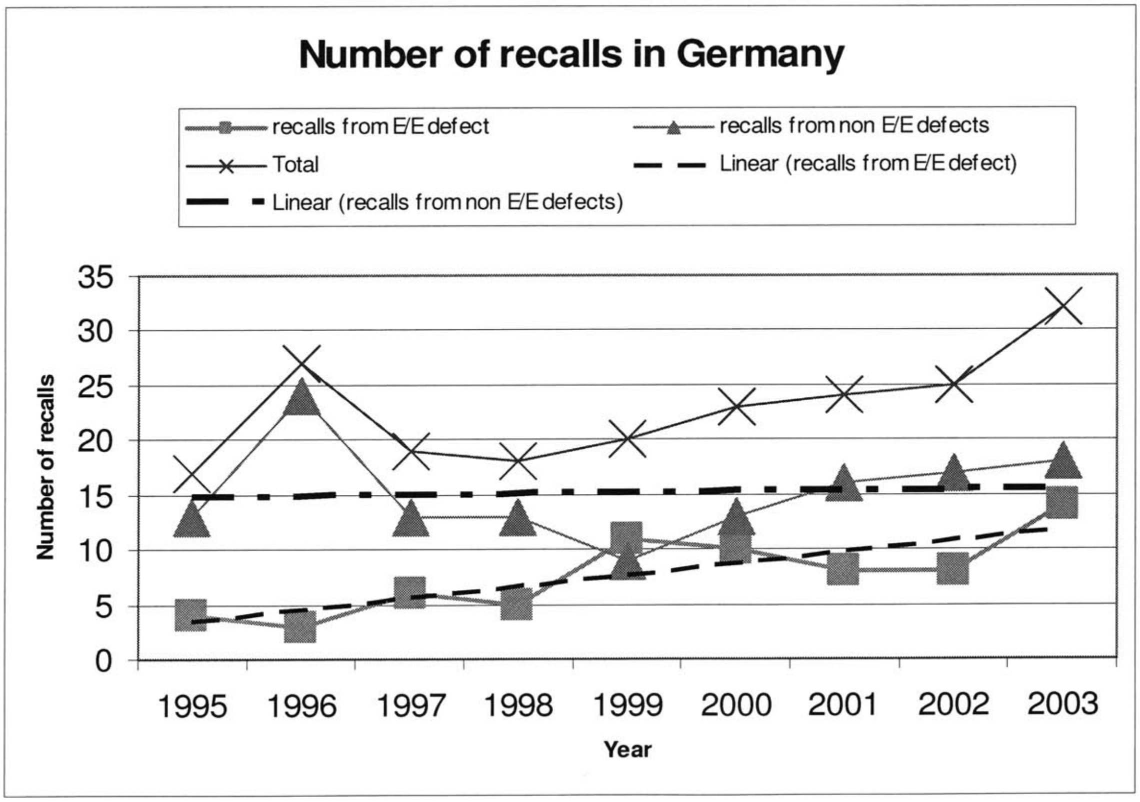


Figure 5. Number of recalls in Germany - Electronics vs. Non-Electronics related

Franz Fehrenbach, chairman of the board of management at Robert Bosch GmbH (Stuttgart, Germany) claims that "[…] there is a direct correlation between the number of electronic functions and the number of defects per vehicle"[2]. As Figure 6 shows, the relationship between the number of defects and the amount of electronic components in a vehicle is indeed approximately linear.

McKinsey points out that half of all electronic problems stem from software defects. But more interestingly, those software-related defects are much more expensive to fix, in particular because it is difficult to find them[13].

As noted earlier, it is not the ECUs themselves that pose a problem but the communication between ECUs, both from an infrastructure and from a software point of view. The interfaces are not standard, and the role of system integrator is thus paramount.

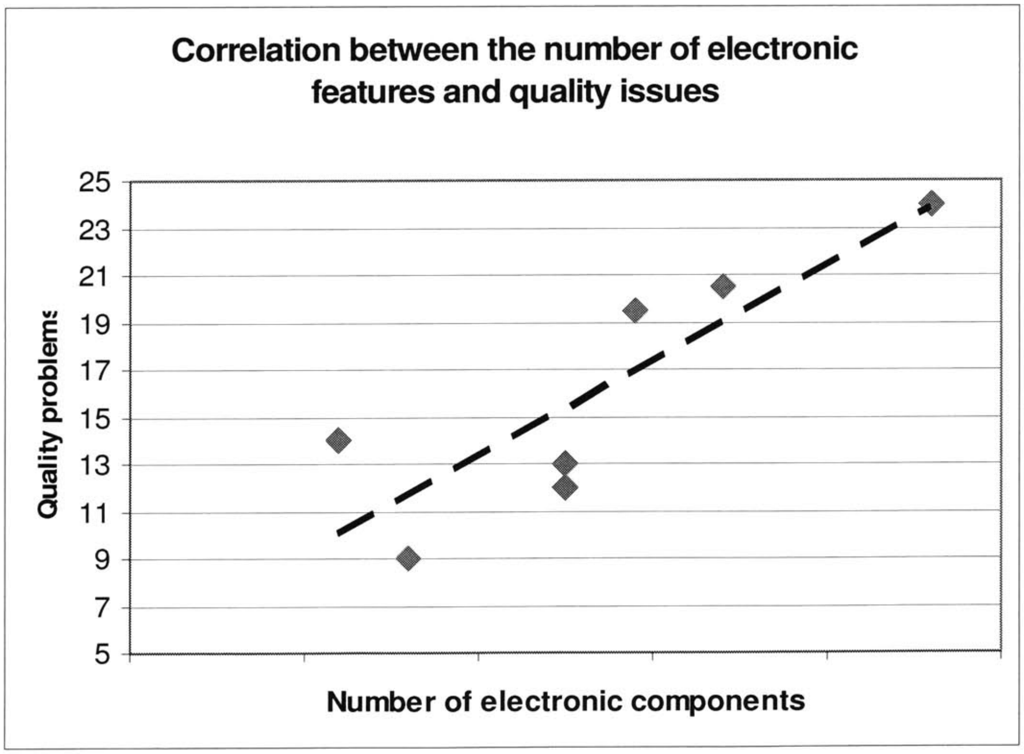


Figure 6. Correlation between electronic features and quality issues[21]

Disconnect between what Customers Want and what OEMs Offer

Interestingly, the proliferation of electronic components in vehicles does not seem to quite correlate with the demand from end users. With the flexibility and customizability offered by electronics, several OEMs have had a tendency of adding functionalities at a breakneck speed, seeking first-to-market advantages, regardless of whether the voice of the customer really guided these choices or not. The difficulty in system integration of those functionalities prevented many failure modes from being discovered and eliminated, leading, as shown in the previous sections, to undeniable quality issues. The irony is that many of those added features do not add any perceived value. For some functionalities in fact, OEMs have completely failed to raise any customer awareness. For example, an OEM offers the ability to close all windows and sunroof when the air circulation button is switched to recycle (to seal the inside of the car from the outside), but most customers were unaware of this functionality. Similarly, customers were unaware of the possibility to program how long the headlights would stay on after the car was turned off.[13]

The quality issues however manifest themselves as failures that customers are definitely aware of, which translates in a negative impact on a brand.

II.B.4.b) Where Does Complexity/Complicatedness Reside in Automotive Electronics?

As we have seen in the previous section, electronics has both the potential to bring value as well as the potential to increase complexity, which, if improperly managed, turns into complicatedness, and thereby destroys value. In order to get the former and not the latter, it is essential to understand where complexity resides so that it can be managed so as to reduce complicatedness.

What transpires from the previous section is that there are two important components to the management of complexity: interfaces and knowledge.

Interfacing

Several automotive OEM and supplier employees were interviewed and all agreed that the integration of modules was the most important driver of complexity.

Eddie Khan mentioned that “since most vehicles are becoming modular, the biggest hurdle is getting each module to respond appropriately to each other on a common network”[22]

Many customers also acknowledged that supplied software components are really black boxes to them. One respondent explained that an OEM must pay special attention to the specifications given to the suppliers and make sure that all functionalities and conditions of utilization are properly planned in advanced because whatever is put on paper if what they will get. Since software products are such black boxes, their integration is more problematic. It is therefore essential for customers to gain expertise in software design, testing and integration. By doing so, a customer is in a position to better decompose its product so that it can “own” and manage the interfaces instead of letting its suppliers control the system and provide a black box.

This will be discussed more in detail in section IV.B.1.

Knowledge Gap

One of the main reasons why communication is such an issue, aside from the lack of standardization in the software domain, is the fact that the competency required is relatively new to the automotive industry.

From an organizational point of view, many manufacturers (OEMs or suppliers) rely on separate electronics and mechanics departments that have their own independent development processes. Historically, the focus has been on the mechanical domain; but with the shift in paradigm to an electronically controlled car, it is essential for OEMs and suppliers to ensure that they obtain the knowledge they need in the domain of electronics and software.

Even if companies hired new talents to create a software-knowledgeable employee base, it is important that those employees be placed at various levels of the corporate hierarchy including purchasing, systems engineering, but also and perhaps most importantly, in management so as to plant the seeds for a good understanding of what is at stake among the decision makers. One interviewee mentioned that managers have not realized that software can not be developed like hardware – in particular, software development is highly iterative and uncertain, and implementation is very much coupled with development.

Alternate sources of knowledge are partnerships with universities and other industries, but according to McKinsey, companies in average do not seem to make much effort to seek knowledge in those places.[13]

II.B.4.c) The Impact of Standards

When asked what role industry standards have played, most interviewees mentioned the effects that standards have on cost. By providing a common tool to all players, standards helps reduce or eliminate a lot of waste; the type of waste that comes in the form of (often late) rework, excessive communication between suppliers and customers, waiting, and as a result, undetected failure modes.

Consequently, by reducing this type of waste, standards give every protagonist an opportunity to put more focus on innovation. Eddie Khan mentioned that "*Industry standards have helped advance electrical architecture for each manufacturer. Things that used to be optional equipment to our customers are now standard, hence making vehicles safer & safer. (e.g. air bags/side air bags)*"[22]

However, along with lowering costs, standards decrease the competitive landscape and lower barriers to entry. Indeed, with increased standardization, the required expertise for integration is lessened, thus lowering the value-added of the integrator role. Mark Schaefer explains that "*electrical vehicle architecture is well known, and the learnings are out in the open. The incentive to roll your own electrical bus architecture is dwarfed by the expense, and most drivers aren't going to pay a dime more for a more advanced architecture that doesn't provide them significant benefit*"[23]

By simplifying interfaces, standards allow a customer to concentrate on thinking up a more elaborate system design at a higher level because it does not have to specify the design of each components, only the interface specifications. Therefore, the design of the whole system consists of thinking up the functionalities that each subsystem ought to have and how they would integrate together.

However, standards can also have a negative effect. If the customer takes advantage of the simplification of interface management only to reduce cost and not to enhance its own system value creation, it may lose competency in the integration of subsystems. In other words, standards should be utilized to shift investment focus from lower level components design – where suppliers can bring the value – to high level system design and development so that whatever value is created from the suppliers can be captured in the whole system. Unless this is done, value gets concentrated in modules from suppliers. Moreover, this encapsulation of value means that potential suppliers can focus their efforts and expertise more specifically and be more easily competitive against incumbents. A closer look at barriers to entry is taken in section V.C.

A trap that companies (particularly OEMs) may fall into is to only see an open standard as "pre-packaged" engineering. An analogy may be drawn with commercial finite element modeling (FEM) software products: in order to make it easy for users to utilize the program, many commercial FEM products have simplified user interfaces to the point where, unless expressly specified otherwise, many parameters are set automatically with default values. As a result, even a novice user can manage to get a result, but he or she may not understand what actually goes on in the background, consequently there is no guarantee as to how meaningful that result is.

The simplification of interfaces, for FEM products as for automotive electronics software control, helps make the achievement of a product easier, but the fundamental logic or mechanism behind it all remains the same and remains necessary. By making it easy to reach a result, it may also prevent users from trying to actually understand what really goes on in the background. That is the danger that OEMs face. It may be easier and easier for them to implement functionalities through electronics, but they need to understand how the system works to ensure that the end product is operational and to detect failure modes.

III. The Effects of Modularization in Other Industries

III.A. The PC industry

As we will see in the following pages, the PC industry has undergone significant transformations as the result of the modularization of its dominant design. We will see through this example how the industry was disrupted and how modularization caused such a transformation. The nature of the car industry is certainly quite different, but because the PC industry is such a clear case of how modularity impacted the interdependencies and relationships between the various players, the following sections will provide some insight as to how the power balance in the auto industry could potentially be affected.

III.A.1. How the PC Industry was Disrupted (or the Power of Modularization)

Baldwin and Clark[24] describe what happened to the PC industry when IBM modularized and opened its architecture. They assert that the explosion or vertical disintegration that it triggered may question the generally accepted dynamic of industries that Utterback has described[25]. At the beginning many companies are created that all bring a new product design to the table, then, when one design emerges as dominant, the rate of product innovation decreases as the rate of process innovations increases and with a stable product design and mature process design, the industry consolidates and the number of companies within the industry falls dramatically.

The story of IBM's architecture modularization and its influence on the PC industry however seems to be a phenomenon of a different nature and thus does not necessarily conflict with Utterback's theory. In one case, it is about the birth of an industry and the generic dynamic that governs its evolution, in the other, it is about a dynamic incident that perturbed a seemingly stable state of an industry, destroyed the equilibrium and forced a reshuffling of companies toward a new industry equilibrium.

Baldwin and Clark show the explosion of the PC through the distribution of market value among PC industry sectors over the years.

Figure 7 shows that until about 1970, most of the computer industry value was held within IBM. In the 60s and 70s, the computer architecture became modular. From 1970 on, new industry sectors slowly started capturing a share of the industry value. Then in the 80s and 90s, more companies entered and new industry sectors aggressively added value to the industry. It is notable that IBM's market value went considerably down in the early 90s as a result of other companies taking over the design and manufacturing of many modules. It is however also notable that IBM's market value came back to a high level in the early 2000s. What is interesting about that is that it is through a new business model that IBM came "back to life" while other companies and industry sectors have emerged and their market value has skyrocketed. Figure 8 shows the aggregate value of all companies in the PC industry. While this shows the creation of what is now commonly referred to as the internet bubble and its burst, with the creation and disappearance of about $3 trillion between 1996 and 2002, we nevertheless observe that value has been redistributed over many companies that restructured into more specific sectors. Indeed, the switch to an architecture that was more modular and more open disturbed an industry that was otherwise in a seemingly stable equilibrium state and thus started a dynamic phenomenon with strong consequences on value creation and value distribution or migration.

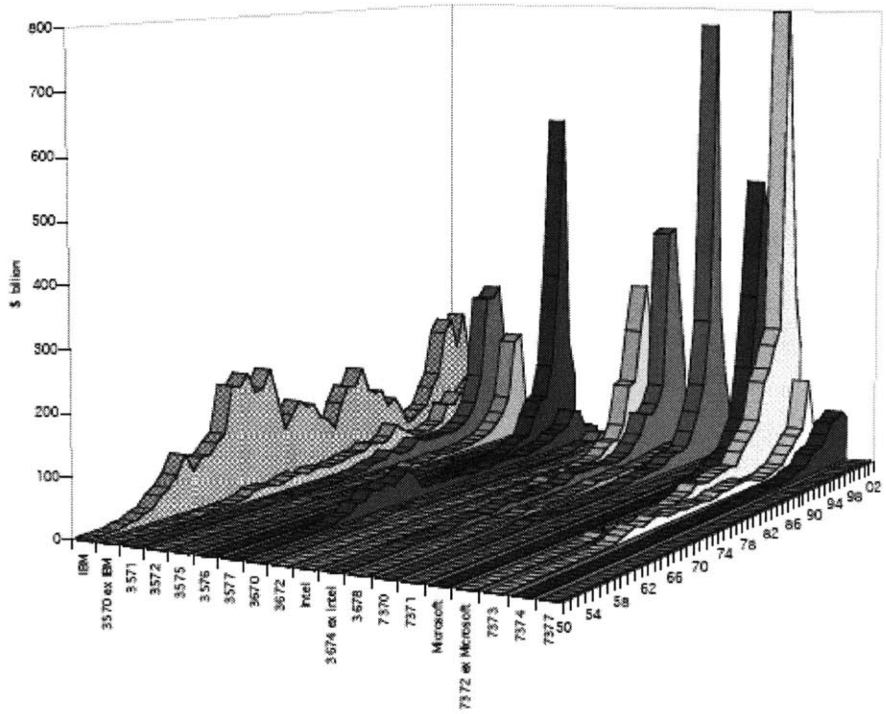


Figure 7. Market value of the US computer industry by sector in constant 2002 dollars (From Baldwin & Clark[24])

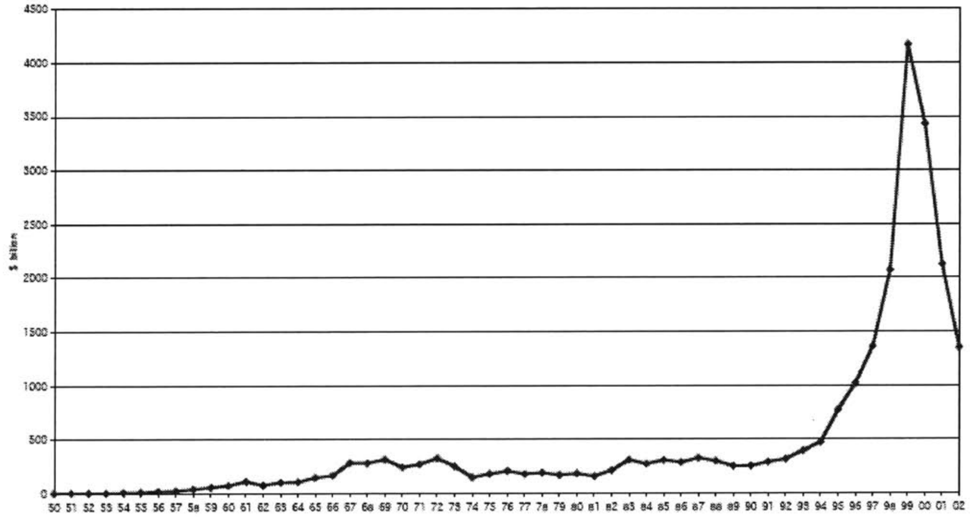


Figure 8. Market value of the computer industry aggregated (in constant 2002 US dollars) – (From Baldwin & Clark[24])

III.A.2. Types of Modularity

Baldwin & Clark have identified 3 different types of modularity:

- Modularity in design
- Modularity in production
- Modularity in use

Modularity-in-use is defined at the functionality level; the customer controls which functions the system should have, is able to purchase modules from a wide range of choices and can therefore assemble a highly customized product. Examples of modular-in-use systems include audio equipment, bedding products, etc.

Modularity-in-production is defined by the ability of a producer to segment the final product into modules that are to be fitted together eventually to form the final product. It is defined at the assembly level and a different manufacturer could not necessarily take a module from the first manufacturer and expect to make it fit in its own design.

Modularity-in-design is the most interesting modularity concept. It is defined by the ability to design the system (not the modules) through the combination of various independent modules.

One may argue that modularity-in-use and modularity-in-design are close concepts since a user mixes and matches modules in order to design his or her own final product, just like a company uses modules to design its final product.

Nevertheless, the interesting point here is that there is often confusion between modularity-in-production and modularity-in-design. Simply breaking down the production of a system into certain modules that can be independently manufactured and then assembled together at a later time is a proposition of a completely different nature than having a system's architecture that only determines the interrelationships between modules in a way that does not limit the value that these modules can bring to the overall system, and hence lets the modules shape the system design.

III.A.3. How Modularity Causes such a Dynamic of Value Creation and Migration

Baldwin & Clark explain that designs have option-like properties. Given certain constraints, many designs may be created, thus the designing process is a process of choosing among several choices or options. For a completely integral design, there is only one choice, the "system option"; however, as the design becomes more modular, the system is defined by its architecture, but its design is determined by the various modules, hence, the options are multiple. The more modular the architecture is, the more options exist, and the more value may be created.

Indeed, rational engineers would select one design over another only if the first one is better, thus, modules are individually optimized, each creating as much value as possible, yet conforming to the design rules specified by the architecture. Baldwin & Clark argue that, because there are more options with a modular design, the probability for the system design to be better, or to have more value, is higher. Moreover, the time-dimension is important as well; while the architecture remains the same, modules may be improved independently; hence, value may be added to a system incrementally.

> *"[...] "modularizing" a system involves specifying its architecture, that is, what its modules are; specifying its interfaces, i.e., how the modules interact; and specifying tests which establish that the modules will work together and how well each module performs its job."*[24]

This sentence from Baldwin & Clark is key because it tells that it is in the architecture of a system that lies the ability to create value. It implicitly says that a good architecture is one that specifies the interactions between modules with as little constraint as possible, thus allowing as much value creation from the modules, but with enough constraint so that the whole system product meets its intended functionality.

They illustrate their point with the Design Structure Matrix (DSM) representation of the modularization of a laptop.

A DSM is a matrix that maps the interdependencies between tasks or components in a project or system. Although it is not shown in Figure 9, columns are the same as rows. A DSM is therefore a square matrix. Reading across a row tells what information or component is required in order to fulfill the task or create the component corresponding to that row.

As we see in Figure 9, although there is an effort to separate the drive system, main board, LCD screen and packaging into separate modules there are many marks away from the diagonal and outside of the module boxes, indicating a dependency between these modules. It is therefore not possible for each module to be designed independently. Figure 9 therefore shows a system that is not truly modular-in-design, although it may be considered modular-in-production because the modules may be produced at separate locations before being assembled together to make the final product.

Figure 10 contrasts with Figure 9 in that there is now a matrix within the matrix where there is no mark outside of the module boxes, indicating no interdependencies amongst them. In order to achieve that, the matrix from Figure 9 has been reorganized and two new elements have been incorporated, namely the design rules, and the system testing and integration. Any dependency that may exist between the modules is now governed by these design rules - that are in effect standards. The key point is that the module design teams do not have to look at each other anymore to find out what the constraints of their own designs are. They only have to look at the design rules, it is helpful in many ways, but particularly because it fixes what the constraints are – i.e. the constraints do not evolve with other modules' design – it centralizes them so that each module team knows what all the constraints are and do not risk to miss any; and it avoids closed loops where module A's design affects module B's, but module B's design also affect module A's.

At first glance, it seems as though this modularization rigidifies the design process, which would appear to be counter to what was argued earlier. Indeed, in order to be effective, the design rules are frozen. Not freezing them would defeat the purpose of what a rule is supposed to accomplish – imagine what would happen if the rules of football changed all the time. But by the same token, the direct interdependencies between modules are removed, and this is where the benefit lies. With the need to obey only a single set of rules, the design space is considerably widened, allowing for more options, hence higher value creation as discussed earlier.

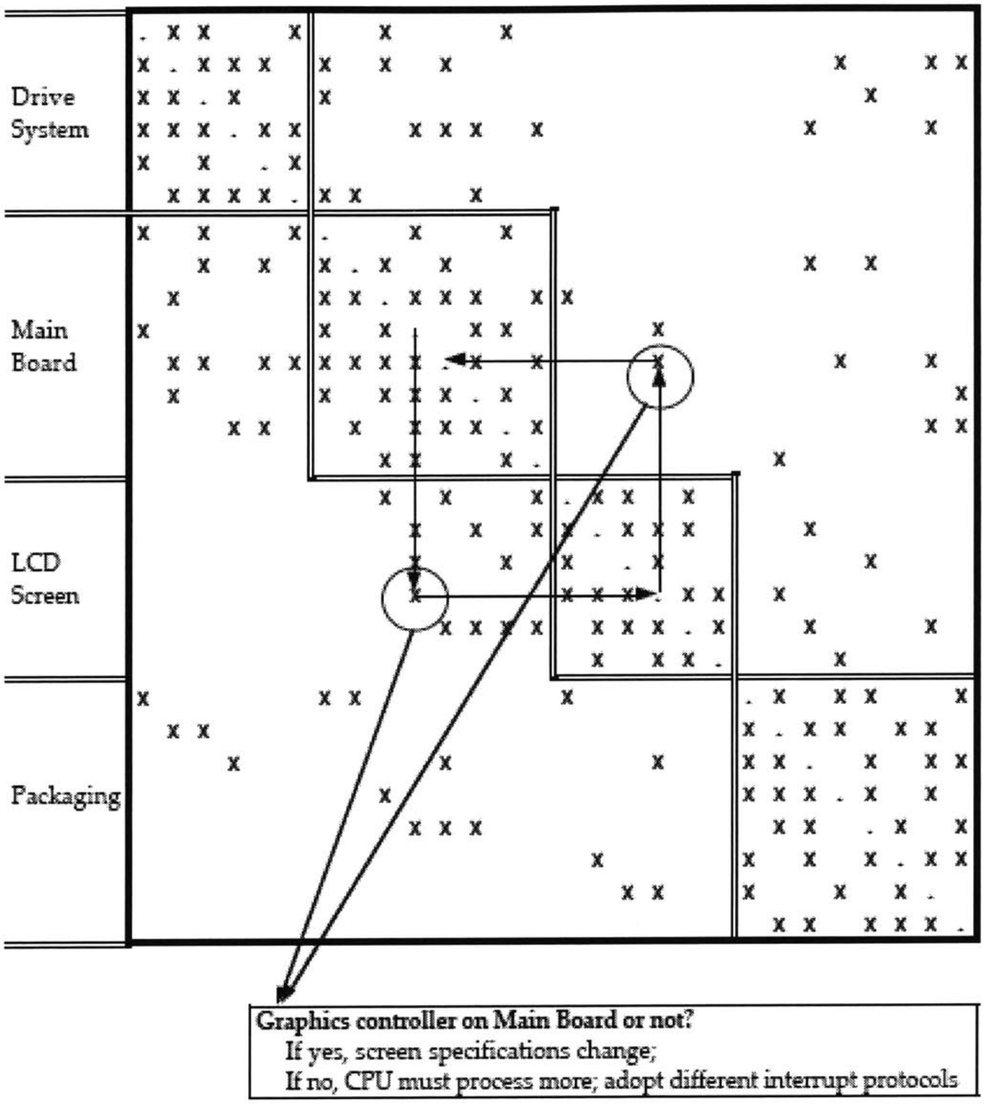
It is possible however to create too many rules that are too rigid, thereby hindering the ability for modules to develop value by limiting the number of their options.

This is why system architecting is an essential competency. There is a compromise to be found between one that is too detailed, allowing a break-down into more modules but putting too many constraints on them, and one that is too high-level allowing more freedom in the design of each module but limiting the number of modules. To put it in terms of value, the optimum has to be found somewhere between more options with potentially less value each, and less options with potentially more value each.

As Fine & Whitney mentioned "*A basic skill of system engineers is thus to assess the "decomposability" of a system and to seek good ways to decompose*"[9]

The DSM is a very good tool to evaluate how to effectively decompose a system into modules, and as illustrated in Figure 10, to determine how to specify the system architecture with the proper design rules.

One can now understand how this ties back to Figure 7. By creating a modular-in-design architecture, IBM allowed various companies to focus on specific modules and develop designs independently of each other. This allowed many companies to bring more value.



Graphics controller on Main Board or not?
If yes, screen specifications change;
If no, CPU must process more; adopt different interrupt protocols

Figure 9. Design Structure Matrix of a Laptop (from Baldwin & Clark[24])

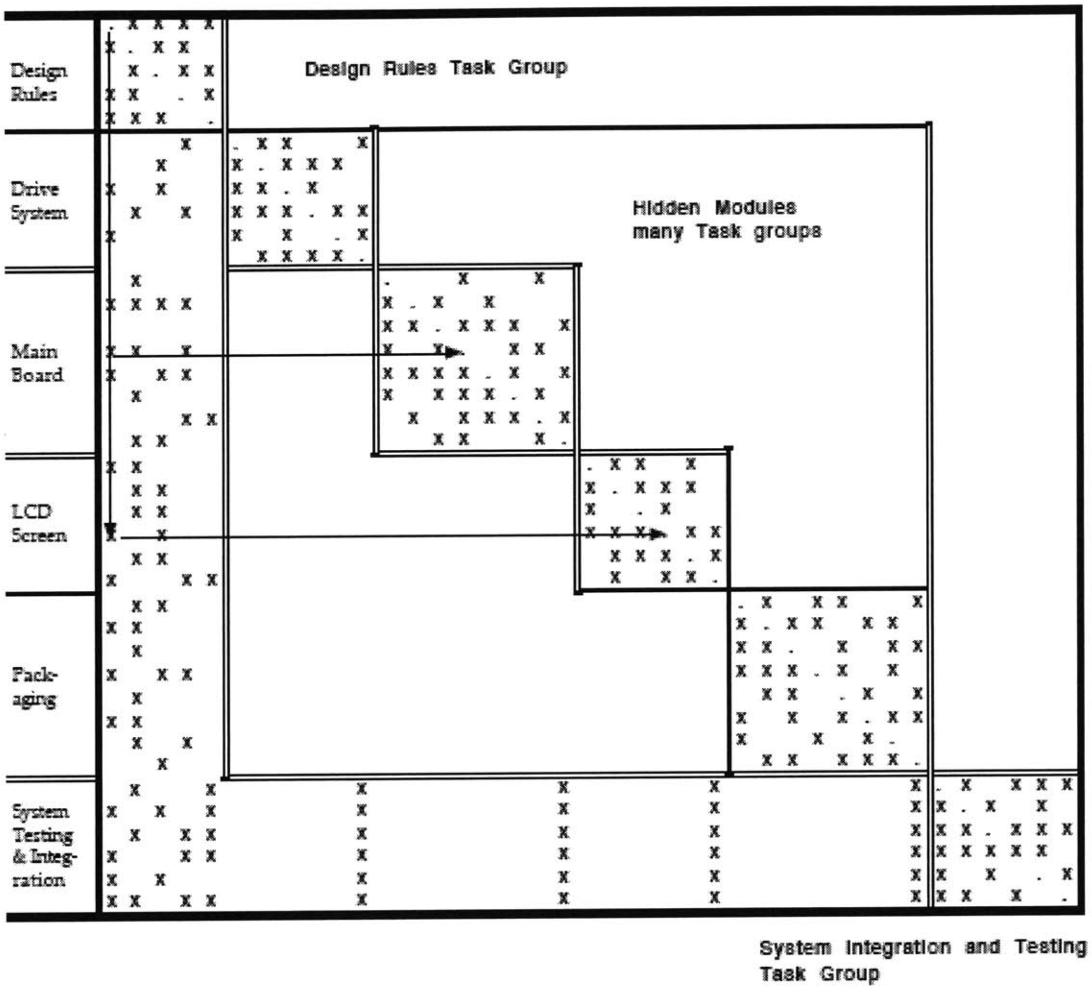


Figure 10. DSM of the "modularized" laptop design (from Baldwin & Clark[24])

III.B. The Aircraft Industry

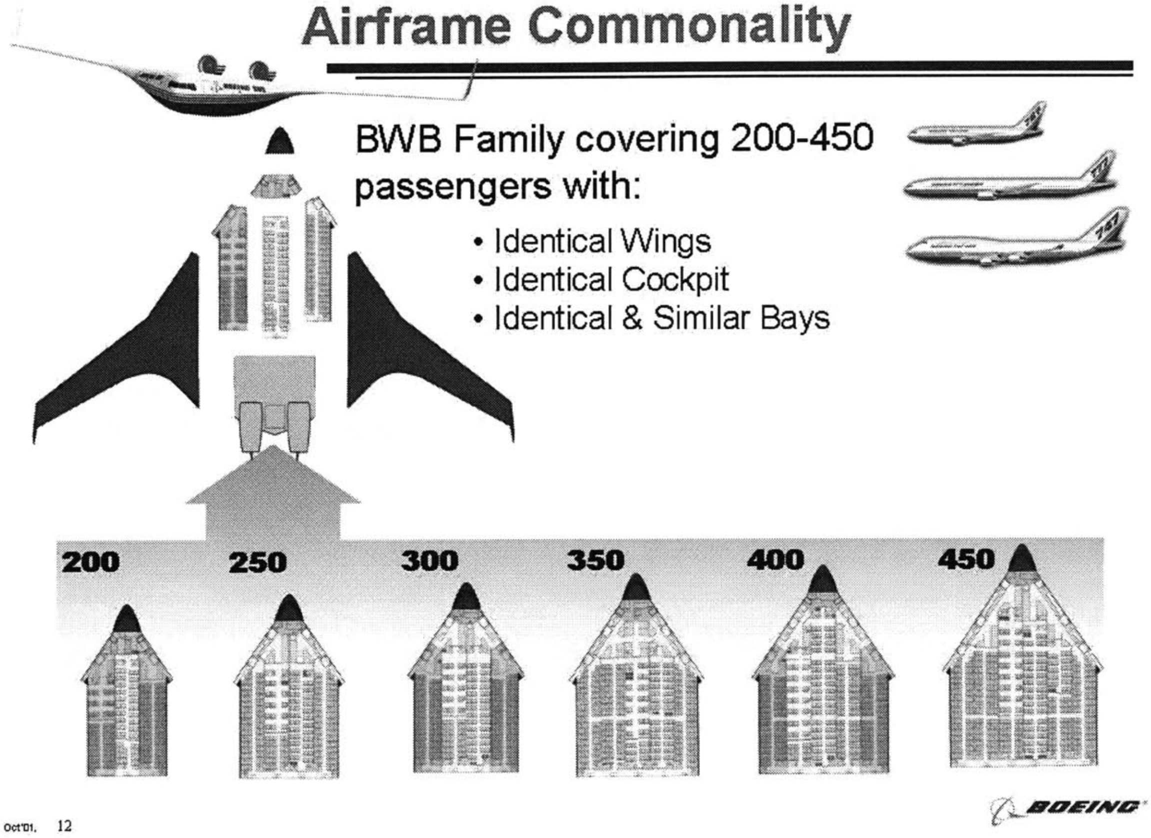
We already mentioned in section II.B.4.a) that the aircraft architecture has been transformed by the advent of fly-by-wire. It is interesting to note the extent to which this transformation has led to the modularization of the architecture, and the impact this had on the industry. Since many technologies utilized in the automotive industry are often pioneered in the aircraft industry, a look at the evolution of airplanes' architecture may provide some insight into what may be expected in the automotive industry while keeping in mind that volumes in both industries are quite different.

III.B.1. Increasing Modularity in the Aircraft Industry

In their book "The Power of Product Platforms", Meyer & Lehnerd explain that Boeing captured value by changing the way they did business and started a new model based on a platform architecture for the 777. The motivation behind such a radical course of actions was the difficulties that the company experienced in having to manage an increasing number of different architectures. This new model called for a new organizational structure within the company: a few core competencies were identified, and the new organization was centered around those competencies. They claim this new business model provided big improvements in terms of efficiency as fewer employees were required to perform similar tasks.[26]

The organizational structure changes extended beyond the boundaries of the newly redefined company. Boeing decided to involve its suppliers in the design process. This became possible because the interfaces between suppliers and customers became more standardized, thereby lowering the cost and complexity of the transactions between them (see IV.B.2).

The example of the blended-wing-body (BWB) airplane concept illustrates further the extension of the modularizability of airplanes (Figure 11). Bob Liebeck, professor at the University of California, Irvine, claims that, as a result of such a design, the number of common parts shared across planes from the same family would represent 39% of the total plane weight, cousin parts would be 33%, while parts unique to each plane would be 28%. Such modularity would not be achievable if the interfaces between modules were not standardized.[27]



Airframe Commonality

BWB Family covering 200-450 passengers with:

- Identical Wings
- Identical Cockpit
- Identical & Similar Bays

Oct'01, 12

Figure 11. Modular architecture of the BWB represents a platform for a family of planes of various sizes (from Liebeck[27])

More specifically, since the advent of fly-by-wire, the avionic infrastructure of aircrafts has become very much standardized and most of the communication is transmitted through a common databus architecture. For example the new Airbus A380 – not yet flying commercially – uses extensively the Integrated Modular Avionics (IMA);

flight controls, entertainment system, navigation systems, etc. all are managed using the same architecture. Moreover, the implementation of the IMA allows and encourages the use of commercial-off-the-shelf (COTS) designs.[28]

As a result of this shift, Airbus claims fewer servers are necessary in the whole plane while more software components can be run on each server. So more functions can be implemented in a single box. Those software modules are in the form of cards that have the same architecture, standard interfaces, but different inputs/outputs. So one card may be running the entertainment, another one may be managing the electrical power distribution, another one the navigation system, etc. [29]

Airbus also asserts that through the implementation of the IMA platform, a saving of 50% of avionics parts is achieved compared to previous architectures on a plane such as the A340-600. It insists on the benefits of encapsulating the functionality in software modules that can be upgraded easily and without interfering with the hardware.[30] Those advantages translate into benefits such as lower weight, lower volume, increased reliability, better maintainability.

As it turns out however, the implementation of such an architecture may not be as successful as it looks in theory. Many suppliers pushed back somewhat on the idea of embedding their products on a single common card. They would argue for example that the card is not adequate for the specific application (sensors interfaces, processor memory requirements, etc.).[31] As a result, the standardization goals behind IMA were only partially achieved as much customization was performed by suppliers whose reluctance may in fact stem from the fear of seeing at least part of their product become commoditized.

Some systems integrator even prefer integral design simply because of the weight advantages that they provide. With a modular architecture, the parts are more disconnected, more connections are required, and those all translate into added weight. In an environment where many man-hours are invested to shave fractions of a pound out of an electronic component, integral designs seem much more attractive.

III.B.2. Toward the All-Electrical Plane...

Several years ago, the implementation of fly-by-wire meant a shift from hydraulic to electric for flight controls. Today, Airbus is planning on extending the electrical "linkages" to the whole aircraft – not just for flight commands. With their More Open Electrical Technologies (MOET) program, Airbus wants to build on the Power Optimized Aircraft (POA) research efforts sponsored by the European Union. It stems from the recognition that there are currently 4 power sources and 3 power networks in an aircraft and the belief that power rationalization is possible – the current 4 sources of power are Electrical (avionics, lights, de-icing, etc.), Hydraulic (flight control actuation, landing gear, braking, etc.), Mechanical (fuel and oil pumps, engine starts, etc.), and Pneumatic (air conditioning, pressurization, etc.). The ultimate goal is an all-electrical aircraft. Indeed, with current fly-by-wire technology for example, commands are transmitted electrically, but the actuation is still hydraulic. Standards for a new electric architecture are being developed and are expected to enable a more versatile power distribution based on generic modules and power network.

III.B.3. Knowledge Perspective

Over the course of the 20th century, planes, just like automobiles, have evolved form purely mechanical machines to much more complex products made up of subsystems in which hydraulic, electrical, electronic and other domains meet. While the first planes were essentially a mechanical structure with an engine, today's planes require a truly multidisciplinary engineering task force to develop. How did the successful suppliers and manufacturers gain the expertise they needed in order to make what is nowadays such a complex piece of machinery?

A look at key suppliers' history help provide some answers. Hamilton Sundstrand for example started as the Sundstrand Adding Machine Company in 1914 on one hand and the Standard Steel Propeller company in 1919 on the other. Standard Steel – predecessor of Hamilton Standard, as its name indicates, was in the business of aircraft propeller early on. Sundstrand got in business with its engineering and manufacturing of adding machines, but quickly diversified into tooling in 1926 and oil pumps, motors and valves in 1935. As Hamilton Standard grew, engineers within the corporation developed expertise in control, testing and in 1949, the company diversified into hydromechanical and electronic fuel control – which will go into service for the first time in 1954. In the same timeframe, Sundstrand develops hydraulically regulated transmission for the B-36 bomber. It will eventually produce electric power and flight actuation systems for aircraft, both military and commercial.

It appears therefore that for about the first three quarter of the 20th century, the aircraft business has grown organically and products were developed and improved as engineers' understanding of the product increased.

While Sundstrand started a series of acquisition in the late 60s for electronics, controls, and heat transfer, the bulk of the acquisition, joint ventures and consolidation wave throughout the aircraft industry – not just Hamilton Sundstrand – occurred during the 1980s and 1990s and even into the 2000s. It seems that the potential for knowledge transfer was great during this period – whether the knowledge was actually transferred and who really took advantage of it is another question. In particular, the fields of electronic, controls, materials are where much of the inter-business activity happened. It is in 1999 that United Technology Corporation (UTC) acquires Sundstrand Corporation and merges it with its Hamilton Standard division.[32] Today, UTC includes among other businesses Pratt & Whiney who specialized in aircraft engines, gas turbines and space propulsion systems, Sikorsky who designs various types of helicopters, other business units such as Otis (elevators, etc.), Carrier (air conditioning), UTC Fire & Security, and UTC Power (particularly fuel cells). Perhaps more importantly, UTC has a research center (UTRC) that supports the various business units. It is a potential bridge between all the business units that works at the crossroads of the key technologies among those units.[33]

From these observations, one may suggest that there was a threshold relative to the complexity of the manufacturing of aircraft above which organic growth could not guarantee that this complexity could be effectively managed. As expertise requirements grew deeper in several different fields, organic growth could not satisfy these requirements.

Software expertise however has been developed organically. As Steve Bresnahan explains, "*Proper software design is also a key part in passing challenging environmental tests like lightning and electromagnetic interference susceptibility. So, it's probably harder to break into the avionics software business than other software industries*".[31] However, with growing sophistication of software products, the software architecture is evolving toward a more layered one. With this approach, companies are increasingly willing to outsource the development of lower layer software components and perhaps even coding and verification of higher-level software layers as long as those are not regarded as core competency. As a result, a company such as Green Hills Software Inc. has experienced a recent tremendous growth by supplying a real-time operating system (RTOS) named "Integrity" and other applications to the aircraft industry as well as the automotive industry and other markets.[34]

It thus would appear that software follows the same pattern as other fields of expertise with a delay. As complexity rises, the need for modularity increases and eventually modularizability does increase, followed by a certain standardization of interfaces (between applications and operating systems for example) thereby opening the door for outsourcing.

III.B.4. Impacts of Modularization on the Supply Chain

There are some clear repercussions on the supply chain as a result of the standardization of interfaces. Because there are fewer types of software, they can be sourced from fewer suppliers as more functions are handled in a more common way (everything on the A380 communicates via Ethernet). The supply chain is therefore reduced in size, but also in depth. Indeed, with an open standard communication architecture, each component can be sourced in a more specific manner, with a more direct connection to the supplier. And since a supplier can provide several of these software applications, the number of suppliers is not increased.

With the A380, Airbus made some particular efforts to increase modularity in the design-chain. They created an IMA Support & Services department that is supposed to be the sole point of contact between A380 customers and the suppliers. This centralization is enabled by the standardization of the infrastructure which assures a relative plug-and-play environment, as well as the fact that the number of hardware and software suppliers on that platform is only thirteen. One of the supply-chain advantages of this approach is the reduction of necessary hardware inventory because much more functionality is embedded into software applications.[30]

The impact of IMA's reliance on more COTS is not quite clear. It may increase commoditization of certain software and/or hardware components. It could potentially shift the competitive edge toward a more specific range of applications. What seems to be apparent however is that the use of COTS generates some debate about it usefulness. Indeed, issues have surfaced regarding the certification and adaptation of COTS for specific aircraft applications. Some experts contend that the necessary amount of reverse-engineering work for certifying existing products makes the process so costly that it defeats the purpose of using cots and that it is in fact more effective to use aerospace specific application software components.

While the infrastructure has been standardized, it does not seem to have been so much the case for software architecture. And as we have seen, although the number of types of software have been reduced, the actual number of software components has in fact risen. That proliferation and the configuration management issues that it engenders are causes for concern for Airbus. For example, Rolls Royce software architecture for engine controls looks nothing like that for other systems used in aircrafts.

It appears therefore that in the aircraft industry, the physical network infrastructure has been greatly improved, standardized and communized so that it ensures a better management of complexity. However, with an increasing amount of functionality residing in the software domain, it appears that the proliferation of software components seems to be where problems reside.

IV. Toward a New Architecture?

In section II.B, we pointed out that complexity and complicatedness reside at the boundaries of systems. Therefore, the ability to perform a system decomposition according to a given strategy is perhaps the most important skill that a company may have. While Whitney argues that mechanical systems – such as a car – are limited in their capacity to be decomposed into modules, it is clear that the automotive industry has been moving toward modular architectures. In most cases, the motivation for this push is the desire to reduce complexity. While it may be successful at reducing the complexity in manufacture, the effect on complexity in design or complexity in use are not obvious. They may even be negative. OEMs hand out more design responsibility to their Tier-1 suppliers, and ask them to provide module solutions – whereby the supplier provides a pre-packaged product ready to install. Those suppliers in turn decompose the system further and outsource some of the components.

The same trend is holding true for electronics. With more and more electronic functionalities added incrementally and in an ad-hoc fashion into the design of a car, various network standards have been developed, ECUs have been added with each additional function and their number has increased to unmanageable numbers. As a result, the industry is slowly warming up to the idea that modularity in electronic systems may be the response to decreasing (or managing) the rising complexity of today's automotive electronic architectures.

As discussed earlier, many of the communication buses protocols were developed by consortia of companies often including OEMs, Tier 1 automotive suppliers and other electronic firms. More recently however, realizing the struggle of integrating software from various suppliers and for different functions into a coherent system, new consortia were formed with the goal of creating an open standard for automotive electronic architecture. AUTOSAR was founded in 2003, EAST-EEA has similar goals, although perhaps more targeted toward the European automotive industry, OSEK is yet another consortium founded in 1993 that focuses on three areas: communication, network management and mainly real-time operating system[35]. It is interesting to note that several companies hedge their bets by participating in several of these consortia on top of developing their own proprietary architecture. For example, BMW is a member of AUTOSAR, EAST-EEA and also developed its own E/E architecture for its latest 1- and 3-series vehicles[36]. Japanese automakers Toyota, Nissan and Honda teamed up on their side in the past few months to form the JASPAR consortium whose goal is similar to and perhaps competing with AUTOSAR[37]. However, Toyota is also involved in AUTOSAR.

IV.A. Open Standard Architectures

IV.A.1. Objective of an Open Standard

With European as well as North American companies involved in its development, AUTOSAR – which stands for AUTomotive Open System ARchitecture – is probably the most important consortium aiming at developing an open standard for automotive electronics architecture.

AUTOSAR was founded by German companies (BMW, Bosch, Continental, DaimlerChrysler, Siemens VDO and Volkswagen) in 2003 but others have joined since – such as French manufacturers Peugeot Citroen, Renault, and North-American ones with Ford and GM (through its Opel branch).

Its vision is "*an improved complexity management of highly integrated E/E architectures through an increased reuse and exchangeability of software modules between OEMs and suppliers*"[38]. At the root of the project is the creation of standardized software interfaces which would have a twofold advantage: allow flexibility and exchangeability of software components, and decouple software from hardware.

The main objective of AUTOSAR is allegedly to provide a platform that facilitates the management of E/E complexity. As indicated in Figure 12, AUTOSAR aims to address all domains of a vehicle decomposed as follows.

- Internal vehicle controls
  - Engine (or Powertrain)
  - Chassis
  - Safety
- Driver and passenger
  - Multimedia/Telematics
  - Body Comfort
  - Man Machine Interface

The open standard would enable communication between every function across those domains thereby eliminating the need for an ECU per function, it would provide a means for OEMs to integrate software modules from various suppliers using the common platform, similarly, it would allow suppliers to easily adapt their modules to suit various customers without having to redevelop them from the ground up.

Perhaps even more importantly, it would actually provide a framework that would give OEMs the opportunity to develop an architecture from the top down. Most OEMs have been struggling trying to integrate every module ECU and get them to communicate with each other across networks. But it is only recently that BMW pioneered the development of an E/E architecture before the design of the car, thereby enabling exchanging and scaling of functions across car series.[36]

"*Cooperate on standards, Compete on implementation*"[39]

Obviously, although participants do encourage the exchangeability of functions and modules across vehicles, suppliers and OEMs, the line is drawn where they believe that competitive advantage lies. The idea is that whatever is considered a commodity and is not potentially value-creating is within the boundaries of the open standard definition; as a result, suppliers and customers alike are free from the hassle of working out the non-value added details – such as communication between ECUs – and can focus on value-added functionalities.

Such a platform also allows OEMs to select or make changes to a function late in the development process, thus providing increased flexibility.

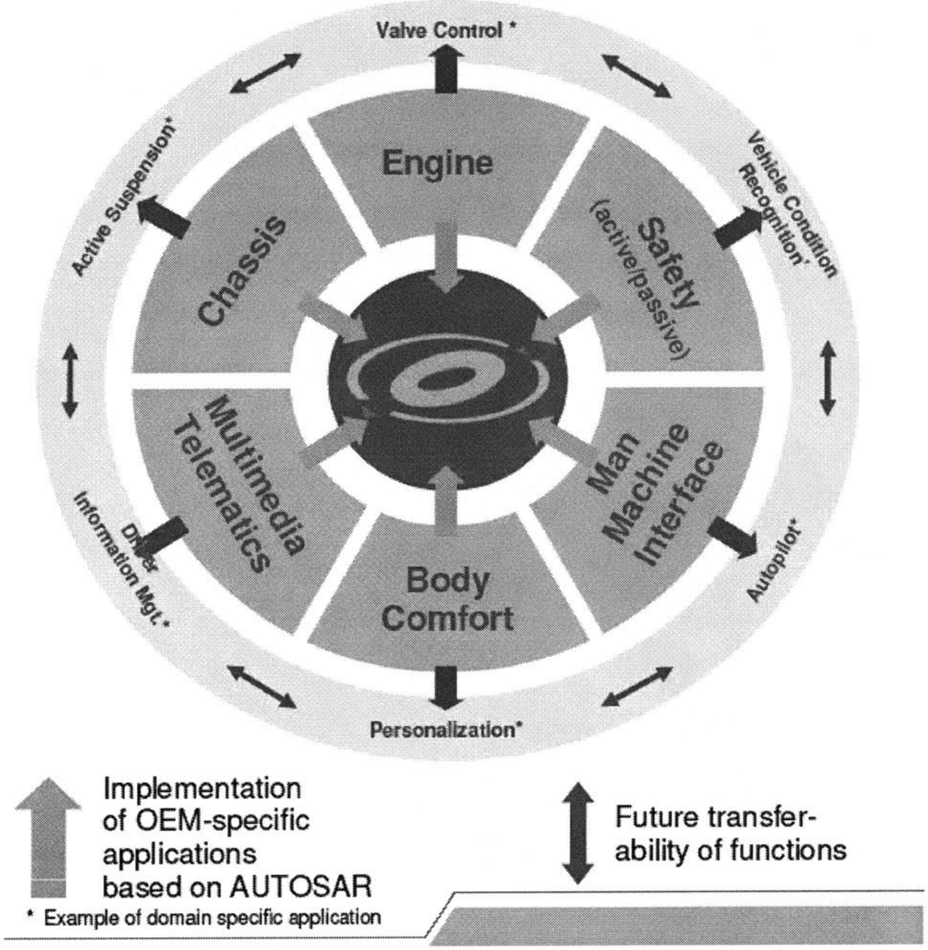


Figure 12. Overview of the AUTOSAR framework (from AUTOSAR official information pack[39])

IV.A.2. Implementation

As stated in the AUTOSAR literature, the four major characteristics of the standard architecture that it proposes are[40]

- Modularity	- Transferability
- Scalability	- Re-usability

Based on these four architectural pillars, the consortium is currently developing a standard that focuses not on specific applications, infrastructure or software, but on the highest level integration of all the major elements of automotive electronics. It embraces other standards that are more specific such as OSEK for real-time operating systems. It supports various standard communication protocols such as CAN, LIN, MOST, etc.[41]

There are three elements to the AUTOSAR architecture

1- A basic software which represents the core and that encompasses operating system, network communication and management services, microcontroller abstraction – which interfaces with the ECU, and ECU specific components such as drivers and ECU abstraction – that decouples ECU hardware from software layers.

2- The Runtime Environment (RTE) that represents the communication environment within and between ECUs. It utilizes standard communication protocols such (CAN, FlexRay, etc.)

3- An application layer that "rides on" the RTE and supports application specific software components.

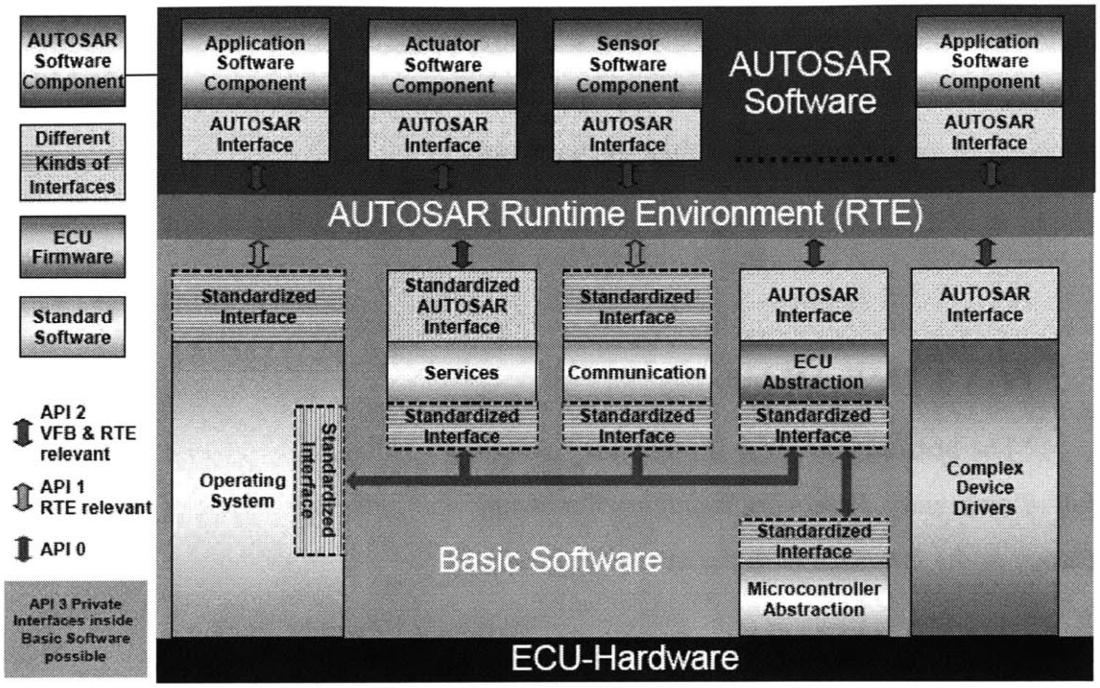


Figure 13. The AUTOSAR software architecture (from AUTOSAR official information pack[39])

Based on this architecture, an AUTOSAR system design consists of the definition of

- its software components that are specific applications that the end-user actually interacts with
- the complete system constraints – description of bus protocols, mapping to ECUs, etc.
- ECU resources – hardware interfaces, connections, sensors, actuators, etc.

One of the key characteristics of an AUTOSAR design is that the application-specific software components are decoupled from all hardware components, type of ECU, type of microcontrollers, network protocols and the location of other software components. Such independence between software components and the infrastructure that it rides on allows great flexibility. It also concentrates the functionality of a certain application in the software rather than the hardware.

IV.A.3. Motivations

"The upstream process is often not "designed" or "run" by anyone - therefore it is full of ambiguity. Removing/Reducing/Resolving ambiguity is a principal role of the architect at the interface with the upstream process"[42]

- Professor Edward Crawley

IV.A.3.a) Complexity Reduction and Cost Reduction as a Driver

The proliferation of electronic components has occurred in an ad-hoc manner. Functionalities have been added in several domains, braking (ABS), transmission, engine management, doors, roofs, telematics, etc. From an organizational point of view, each group responsible for those domains has integrated electronics into their sub-systems. But there has not been any particular entity in charge of ensuring communicability and compatibility among those sub-systems – an E/E architecture, and the integration of this architecture into the overall vehicle system.

As we noted in section II.B.1, complexity and complicatedness reside at the boundaries. By introducing electronics into already existing sub-systems of the vehicle and adding electronic components, a new design layer was added. The problem lies in the fact that the boundaries at this new layer were not completely defined. Only the boundary between that layer and the mechanical sub-systems were defined, but not those boundaries between that layer's components, somewhat like silos. Such architecture would be fine if those electronic layer sub-systems did not have to communicate with each other. This is however not the case. As functionality migrated toward that layer, communication was not only important but necessary.

One may argue that an open standard such as AUTOSAR would partly play the role of "system architect" at the industry level – which, as we noted in II.B.4.c) is potentially a dangerous thing to believe. At least it provides the tools necessary for system architecting within firms by providing a framework for bridging the elements of the electronic layer in a systemic context.

By doing so, it drives away the ambiguity that has built up as a result of the uncontrolled proliferation of electronic components.

This is in many ways similar to what IBM did in the 60s[i] and the consequences it had in the 70s and 80s. The complexity of the computer design was increasingly becoming unmanageable and architects realized that this complexity could be reduced by creating a modular architecture.

A key incentive was also the reduction of cost. If modules are to be supplied by various suppliers, then boundaries between those modules eventually become standardized. Standards in turn allow economies of scale. As Mark Schaefer pointed out, "you can count on microcontrollers (μC) that have a CAN protocol interface". Indeed, this allows the OEM to utilize a generic μC for enough applications so that the volume is large enough for the μC manufacturer to go through the process of environmental certification. By the same token, such standardization is beneficial for suppliers as well because various control modules from multiple manufacturers can utilize the same part number.[23]

---

[i] Although we are comparing IBM – a company – to the auto industry, that company was so dominant in the 50s and 60s that most of the PC industry was concentrated in it as Figure 7 shows

IV.A.3.b) Enabling Mass Customization

Economies of scale are not the only driver however. There are also economies of scope. As an interviewed OEM employee mentioned, one of the biggest hurdles is global architecture. He gave the example of a small car platform manufactured in country A but sold in several other countries. For several outsourcing and marketing reasons, the electronic features that go into the vehicle in country A are not the same ones that go into the vehicle sold in country B. The potential solutions are either to perform the integration of electronic features for each market separately, or to avoid mass customization and utilize the same electronic components for every market.

In the latter case, the outsourcing costs are high because proper marketing strategy cannot be implemented and profits cannot be maximized. In the former case however, the extent of re-integration for each market depends on how exchangeable, re-usable the functionalities are, and how flexible the boundaries are – that is, how easily a component can be replaced by a similar one with different specifications so it can more accurately target a certain market.

While an open standard does not address company specific strategic issues that affect the advantages that can be gained from mass-customization – such as design and production process architecting – it is a platform that enables the management of the complexity that is inherent in a mass-customizable product by enabling the standardization of processes, reduction of design risk and cost[43].

IV.B. Modularization of the Electronic Architecture

IV.B.1. Decoupling between Software and Hardware

As Whitney suggests, there is a limit to modularity in mechanical systems. But it is undeniable that the paradigm of a car as a purely mechanical system is shifting. Some refer to their vehicle as their office on wheels, others consider it an extension of their personal space, yet others see it more and more as a mobile entertainment center.

We argue here that those functionalities are not just ancillary characteristics anymore; rather they define what a car is. A car is not a mechanical machine with some electronic components on it; it is a truly mechanical-electronic system. As Fernando Cela Diaz mentions, such a technology as "drive-by-wire has the potential to make a car look more like a computer"[44].

Given the trend of consortia being created to develop new standards, it is likely that one of those standards will emerge as dominant and widely used for the management of the automotive E/E architecture – be it AUTOSAR or another. The electronic architecture "wants" to be modular.

The evolution of the modularization of distribution of functionality from the human interface to the actuation is described in Figure 14. The human interface is usually mechanical; it may be pressing a button, pushing a lever, applying pressure on a pedal, etc. The stages that exist between the action of the driver or passenger and the actuation of the desired function are broken down as follows:

- <u>Processing</u>: any incoming signal must be decoded or understood; such signals may come from the driver or passenger, or from some other source in the vehicle.

- <u>Functionality infusion</u>: this is the most abstract element. Basically, functionality other than the direct high level intent from the operator may be built into the system.

- <u>Transfer</u>: Signals (informational and energetical) are transferred. These may be any type of signal, raw, processed or infused with functionality.

Note that these are not necessarily in series, i.e. all those stages may be concurrent.

What is notable in Figure 14 is that, as the architecture of the vehicle evolved, the various stages went from being embedded and blurred into the mechanical components to being distributed into specific “carriers”, making it easy to separate each of them individually.

For example, as it currently stands, most electronic components come with software integrated with it. As a result, the whole hardware-software package module needs to be integrated and consequently, it either reduces the customizability or it makes it more difficult. An example given by Mark Schaefer illustrates this problem. He explained that two cars that are built on the same platforms share many modules. More precisely, they share the exact same hardware; it would seem logical that a shared module would have the same part number then. But the software is where the differentiation between the two vehicle modules lies. Because the software is integrated by the supplier in the whole “packaged” module, the two modules need to be differentiated by different part numbers. Moreover, even on a single vehicle, because of the various options offered, the possible configurations are endless – this is particularly true for North American manufacturers, while Japanese manufacturers tend to offer option packages thereby reducing significantly the number of potential configurations – and the need to differentiate between components translates into more tasks and transfers because the software is not independent enough from the hardware or even the electronic. But as the industry realizes this phenomenon, the E/E architecture is shifting from a state that corresponds to the third column from the left on Figure 14 to the last column.

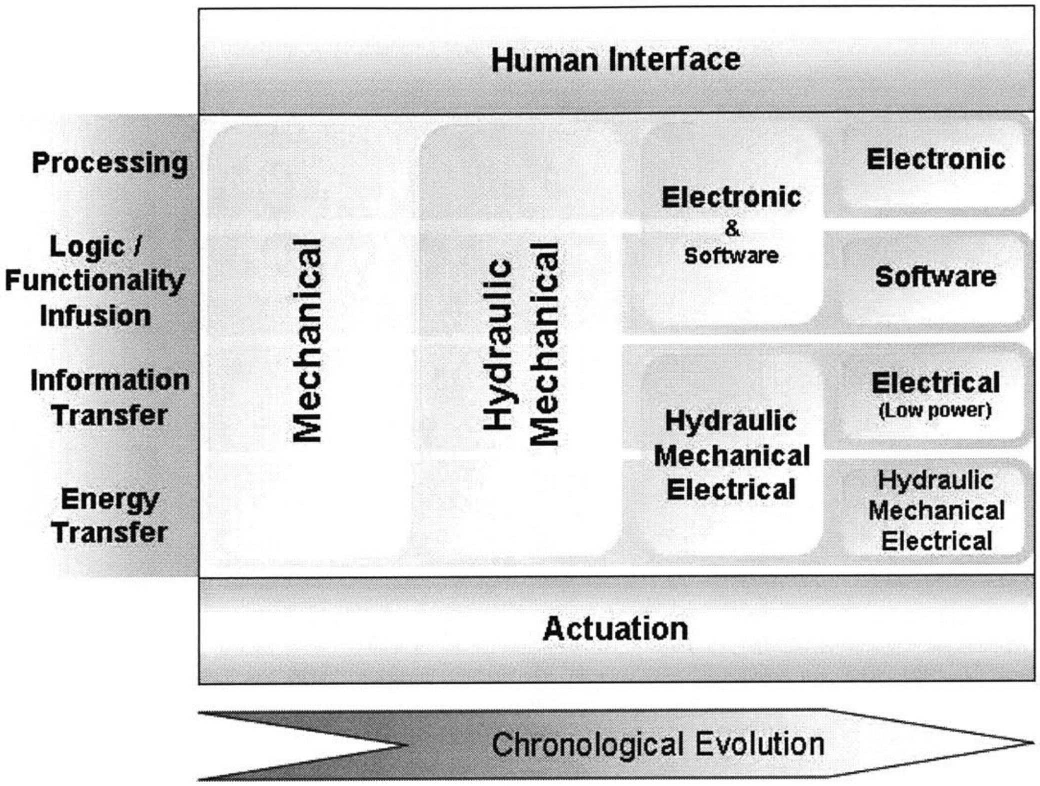


Figure 14. Design modularization and encapsulation of functionalities

The following sections look at two essential functions of the car, steering and braking, and analyze how these functions have evolved.

Steering

In the steering function, the human interface occurs at the steering wheel. The angle that the driver puts on the wheel is a signal. Originally, this signal was processed purely mechanically: the angle of the steering wheel was simply translated into the same angle for the steering column. The transfer of the signal was also mechanical: the angle that the driver puts on the wheel would be transferred through the column, would be translated into a lateral motion via the rack and pinion, etc. and would eventually correspond to a certain angle on the wheels; similarly, the resistance energy of the wheels on the road followed the same route in the opposite way to translate into a resistance against the driver's input. Functionality infusion in this case is more difficult to discern, but for example, a mechanical system that puts a camber angle on the wheel when they turn may be considered a functionality infusion.

The steering function is nowadays being handled, on some high-end cars, by steer-by-wire technology. In this case, the processing is performed by an electronic sensor that detects the angular position of the wheel. The transfer is electrical as well as mechanical. It is easy in this case to see where the functionality infusion would come from. The blue box at the center of Figure 15 is the controller and it can support virtually any added functionality. For example, with its Active Front Steering (AFS) system, BMW made the steering transmission ratio electronically variable as a function of speed and even driving style. As a result, low-speed cornering (to enter or exit a parking spot for example) requires little movement of the steering wheel, whereas at high speed when no sharp turning is desired, it takes more angular movement on the steering wheel to actually move the front wheels.[45]

Note that, as a result of this encapsulation of the functionality, because it is important for the driver to be "connected" to the road, a feedback mechanism is required to translate the resistance of the road back to the driver. However, such a feedback may be customized, it may also be filtered, similarly to how fly-by-wire allowed customization of certain safety function of design envelope on aircrafts.

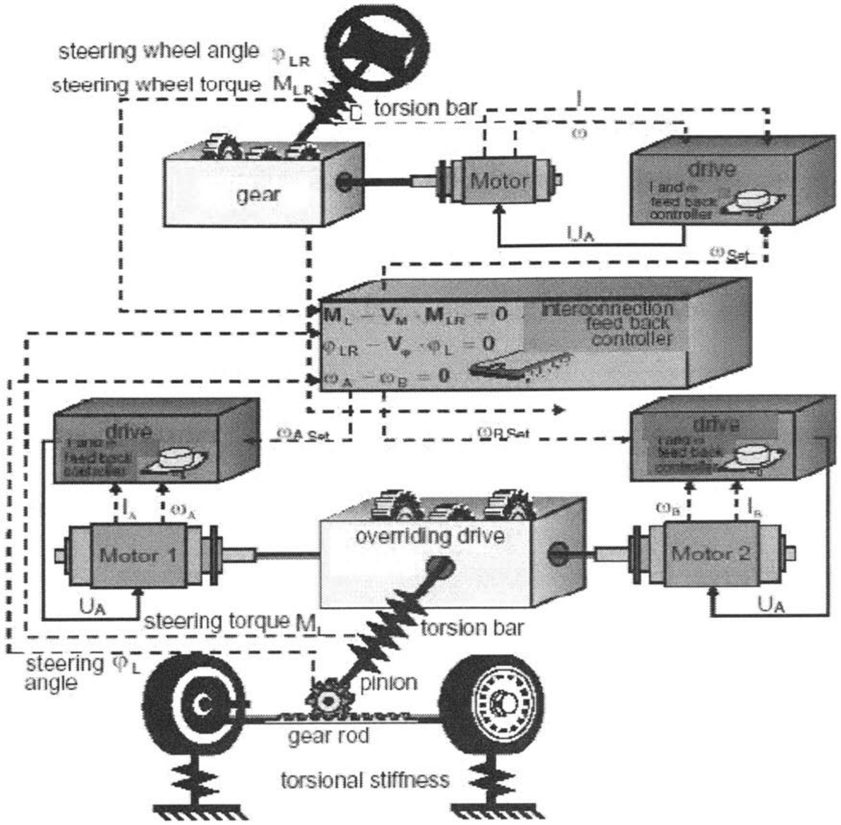


Figure 15. Steer-by-wire system description (from Asam[46])

Braking

For the braking function, an Object-Process-Methodology (OPM) mapping was developed. The OPM is a tool that allows to describe an architecture by communicating its intent, the beneficiary and its need(s), the operand and its beneficial attribute to the beneficiary, value delivery through the operating process, its functional decomposition, form decomposition and the mapping from function to form. It is not the intent of this work to describe the OPM in detail and we refer the reader to the literature for more information on this subject (see [47]).

Figure 16 represents the OPM of a traditional hydraulic brake system. The high-level operating process is to dissipate energy through heat with the intent to slow the speed (beneficiary attribute) of the vehicle (operand). The beneficiaries, not represented there are the driver, passenger, other vehicles and pedestrians.

Given that a process at a certain level becomes an intent at the lower level, zooming on the high-level (Level 0) process yields a decomposition into the following intents (Level 1):

- Dissipate heat
- Transform kinetic energy into heat
- Translate driver input into force
- Transfer input from driver
- Read input from driver
- Feedback

The process to dissipate heat is to move the heated air. The intent to transform kinetic energy into heat translates into two processes: the application of a normal force and the friction between a stationary and a rotary part. The intent to translate the driver input into force requires three processes: force distribution (between front and back wheels), force transfer (to the wheels), force multiplication (to reduce the amount of force required by the driver). The transfer of the driver input requires three processes as well, two of them however are shared with the intent to translate the driver input into force, those two are force transfer and force multiplication, and the additional one is timing (between the front and back if both drum and disc brakes are used). The intent "read input from driver" also maps to the process "force multiplication" as well as pressure sensing from the foot. Finally, the intent of feedback maps to most processes except timing and move heated air.

Each process in turn is supported by a set of objects as mapped on Figure 16, thereby defining the architecture of the system.

Figure 17 on the other hand is an OPM mapping of a brake-by-wire system. The interesting thing between Figure 16 and Figure 17 is where the similarities and the differences are. First of all, obviously, the intents are the same. But the processes mapping to the heat dissipation and transformation of kinetic energy into heat are also the same. Indeed, these correspond to the actuation, the actual mechanism that most directly relates to the high-level intent. It corresponds to a slave component that performs its function when it receives a certain signal.

It is how the signal is transmitted that differs. Indeed the other intents and processes, those that connect the human input to the actuation are different. In the brake-by-wire system, there are two similar processes – mechanical to electrical and electrical to mechanical conversion – that may be considered as the gates between the mechanical domain and the electrical domain. Sandwiched in between are the transfer of the electrical signal and the infusion of functionality.

The interesting point here is that what is transferred is the informational part of the driver input, not the energetical part – this is why a feedback actuator is necessary. The informational signal travels between the human interface and the actuation and the electrical power is delivered independently at both locations.

To better illustrate this, consider Figure 18 and Figure 19. The dotted lines delimitate the part of the architecture that differs between both systems. The mechanical linkage that exists in the traditional hydraulic brake no longer exists in the brake-by-wire system. While energy is obviously still required and is in the form of electricity instead of hydraulic or mechanical, it is clearly more independent from the informational signal. Moreover, Figure 19 clearly shows the separation between the infrastructure and the software – at the boundary of which are the ECUs (there are two ECUs because of the necessary redundancy for such safety-critical applications).

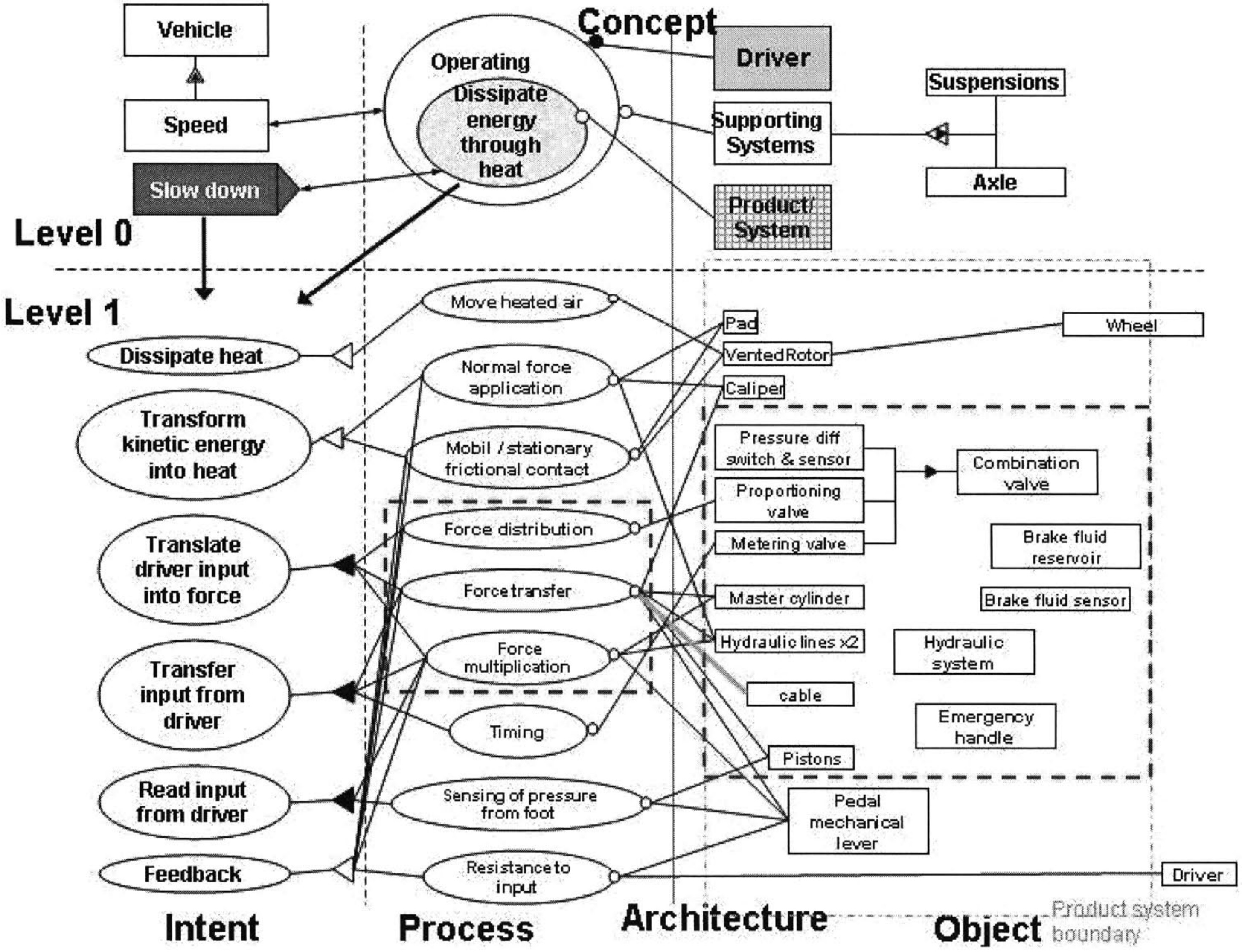


Figure 16. OPM of a traditional hydraulic brake system

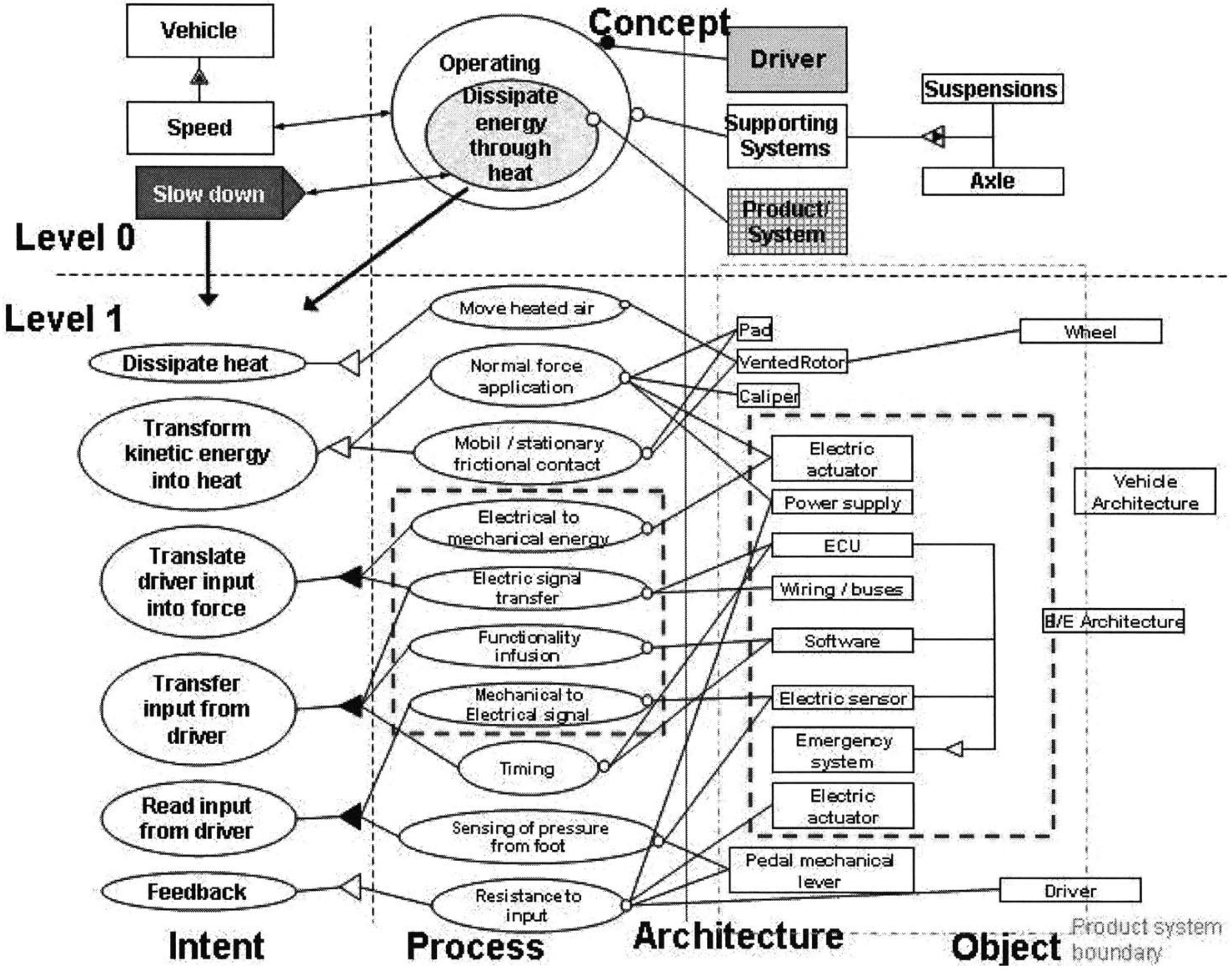


Figure 17. OPM of a brake-by-wire system

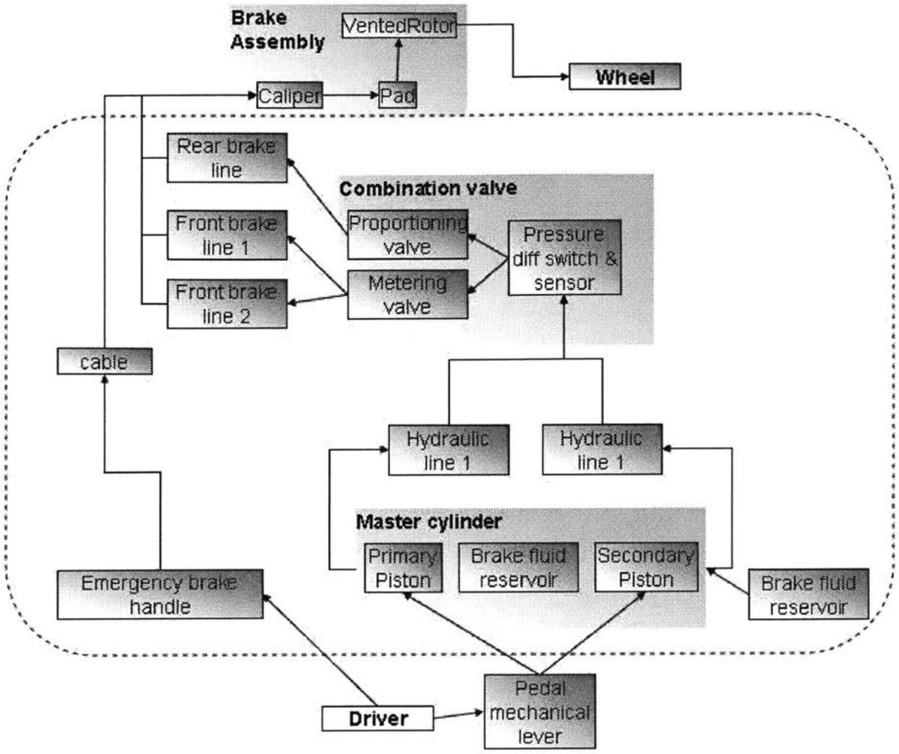


Figure 18. Form decomposition of a traditional hydraulic brake system

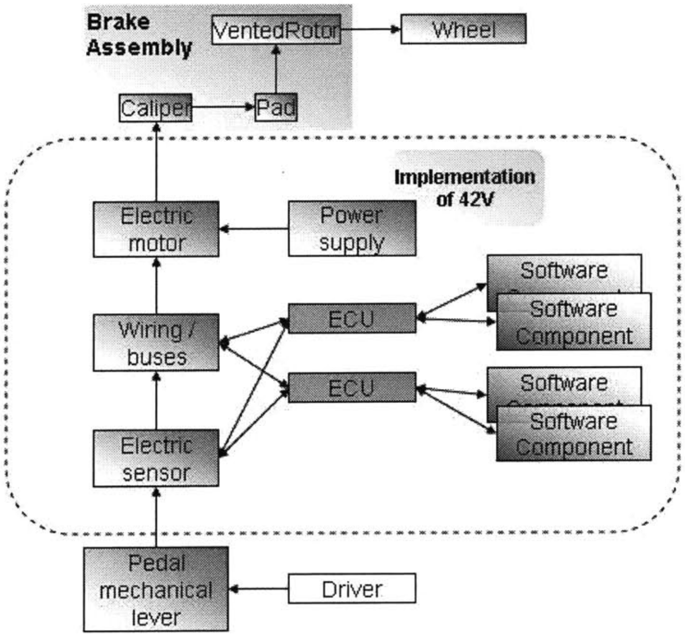


Figure 19. Form decomposition of a brake-by-wire system

An example of the encapsulation of functions is that of GM using its GMLAN (a proprietary protocol built on top of CAN) to connect several functions. When the ignition key is inserted and turned for example, what happens is that it sends a signal to the ECU that the engine should be started; that ECU communicates with the ignition module and power is utilized to start the engine. As a result, adding an option such as "remote engine start" does not necessitate adding any hardware. All that is necessary is to switch on a program in the remote that performs the same action as what the ignition key does.

Discussion

The electronic architecture "wants" to become modular, mainly for cost and complexity management purposes. The move toward more modularity tends to decouple software from hardware. This decoupling is the materialization of the separation between the informational and the energetical signals.

Whitney explains that CEMO systems have a tendency to be more integral in design because their boundaries cannot be described a priori with exactitude, they are not simple logical boundaries and many interactions between modules at the interfaces are not desired. The amount of power is the driver for such integrality. It is this power and the inevitable inherent back-loading that prevents the interfaces between sub-systems to be simple, logic-driven and predictable; it is this power that brings complexity into the system.

The last few sections are therefore worthy of note, because it is ascertained there that the automobile is becoming less of a purely mechanical system, but also that the interfaces are becoming much more distinct between the mechanical domain, the electronic domain and the software domain. What then does this mean to the "modularizability" of a car?

Certainly, a car is still a very complex system, and it will probably remain so, for the simple fact that it requires power to be generated, imposing thereby blurring interfaces with undesired interactions. Because of this, there will always be a certain degree of integrality inherent to a car. But the advent of electronics, with its effect on the isolation of the mechanical domain, is likely to help bring that integrality floor down. And open standards will probably accelerate this trend.

The ESD architecture committee at MIT wrote a generic paper on architecture in which they mention the notion of "*essential complexity*". The essential complexity is "a theoretical lower bound to the complexity of a system"[48]. The proposed principle is that "*Robust functionality drives essential complexity*"[48]. It suggests that, given an intent for value-delivery, there is an absolute minimum level of complexity that can be achieved, and there is a certain architecture at that level of complexity that satisfies that intent.

A solution-neutral statement of what a car is supposed to accomplish is "to move people in an autonomous fashion, while protecting them from the outside environment and entertaining them". When described with such a solution-neutral statement, according to the principle of essential complexity, there is a minimum bound of complexity for a car.

It is argued here that by enabling the separation of the mechanical, electronic and software domains, the standardization of the electronic architecture brings the overall car architecture closer to the essential complexity floor.

The Hy-wire – a drive-by-wire vehicle concept designed by General Motors – is an example that tends to support this argument. Put simply, it makes the experience of driving a car similar to playing a video game. All the controls are electronic and relay the information to the proper actuators and motors. The power is electric, provided by fuel cells. The illustration of the separation of the mechanical domain is made most apparent by the "skateboard" concept, an 11-inch thick aluminum chassis that houses all moving parts.

Figure 20. The GM Hy-Wire concept (from General Motors[49])

Modularization is possible to the extent that a cabin is a completely separate subsystem. In fact, one argument about this vehicle is that the whole cabin can be changed from a van to a sports car. It simplifies the customization such as left vs. right hand driving since the driving controls may be located virtually anywhere. It also provides a great deal of flexibility for the design of the cabin as anything above the "skateboard" chassis is dedicated to housing the driver and passengers – as can be observed on Figure 20, there is nothing between the driver and the windshield that extends all the way down to the "skateboard". Although this is only a concept and it is not mature yet for mass production, it illustrates the fact that electronics, by enabling a shift toward more separation between mechanical components, allows increased modularity and as a result, a better management of complexity, leading to a potential reduction of complicatedness.

IV.B.2. The Design-Chain Perspective

"*You should never lose control of the controls*"[50]

- William Taylor III

Bill Taylor, a System Design & Management alum, introduced the notion of *design-chains*. He defines it as "*the chain of organizations which is responsible for designing and developing a product*". This relates to the relationship that exist between a supplier and a customer relative to the design of a system and how design information is transferred from one to the other, as opposed to the transfer of material and manufactured goods between the two[51].

He explains that for pure commodity product where the design is known and standardized throughout the industry and there is no value within the design itself, the relationship between a supplier and a customer is simple because the only parameter that comes into play is the price of the product. Both functions and forms are widely accepted and recognized as the norm. A customer requests quotes, suppliers bid, the customer chooses the supplier with the lowest price. In a complex system however, the system product design is shared between the customer and its suppliers. But the customer may have a specific idea and understanding of the desired function while the supplier may have its own understanding of the desired function. As a result, the bidding process is not so clearly defined any more. What typically happens is that the customer submits a request with a set of specifications and maximum price, given the complex nature of the sub-system that the customer wants to outsource, the supplier may be able to match the price but only at the condition that the design envelope be somewhat pushed. In this case, the transaction between the supplier and the customer is not solely based on price any more but involves more design parameters. The transaction is therefore itself more complex.

In fact, "transaction" may not be quite the appropriate word. According to Baldwin & Clark, a transaction is a transfer that is standardized, counted and compensated[52]. In a complex system where design is shared across corporations, it would appear that the transfers are only partially transactions. Indeed, a transfer may start as a transaction whereby the customer provides a set of specifications which are {parameter; value} couples. The parameters are a standard format understood by both parties, and the values attached to each parameter is a quantification; finally, there is a valuation by both party of what the service is worth and a promise of remuneration from the customer to the supplier. This does indeed meets Baldwin & Clark's criteria of a transaction. However, due to the complex nature of the system, both parties' understanding of what the sub-system product function is supposed to be is likely to differ to some extent. The transfer that is believed to be standardized is in fact not so standardized. As a result, other subsequent non-transactional transfers occur.

The original transaction was only a transaction in appearance - one may say an illusion of transaction - because there are many unknowns that simply unfold as the transfers between both parties take place. Effectively, as failure modes are discovered, certain parameters that were not even mentioned originally become parts of the specifications. However, no re-valuation for compensation is performed. The alterations to the specifications may come from the customer who may realize that the sub-system product as they originally specified it does not integrate properly in their system, or from the supplier who realizes that they cannot deliver what they promised.

Both Taylor and Baldwin & Clark reach similar conclusions. At the modular boundaries of the design chain, transactions are simple and cost effective because the transfers are already standardized. However, at an integral boundary in the design chain – the interface between a customer, its suppliers, and possibly between the suppliers as well for a complex integral sub-system – transactions are not well defined, they are expensive because complex and complicated.

Most companies (and especially suppliers who can fall into the "hold up" scheme of customers) try to move away from an integral design chain structure. Evidently, an open standard would accelerate this trend.

Specifically, in the case of the automotive electronic architecture, as we have seen in the previous section, an open standard would tend to separate the mechanical, electronic and software domains. Although we have mentioned in section II.B.4.a) that functionality and with it control reside in software, it is in fact often the case with an integral design that a supplier is forced to share its software code with the customer if they want to do business with an OEM. In effect, a supplier may provide a sub-system with a piece of software code attached to it. This code is supplied as a black box. But the customer who tries to integrate the product into its system may have issues because the boundaries between the supplied sub-system and the OEM system are not fully standardized. Because the OEM is not able to communicate well with the supplier software, it will typically request the software code to be shared. At that point, the supplier has little choice because it has already invested time, effort and money into the development of the sub-system. Clearly, this is not a situation that a supplier wants because this means giving up an important piece of intellectual property and the control of a design.

Such scenarios happen because the communication tools are not universal. As an illustration, when two people discuss, they exchange sequences of words. The words themselves or their sequence may sometimes be interpreted differently, and although the people seem to listen to each other, and seem to understand each other at first, it is frequently the case that they actually have to get back with each other to clarify and agree on their understanding of their initial exchange. Very similarly, when two subsystems controlled by software codes try to communicate with each other, they may use the same protocols, but the algorithms may have some incompatibilities that will eventually emerge during the integration process.

An open standard that encapsulates the software component independently of hardware and other software components would prevent such a scenario to happen. In order for an OEM to integrate the sub-system, the supplier would no longer need to provide its software because the transfer would be much more standardized and the function delivery would be simplified.

Controlling the controls is key to survival in business. Since software is the quintessence of control in the E/E architecture, controlling the software is crucial and therefore, through the encapsulation of software components, the standardization of the

E/E architecture may determine who will win and who will lose according to who owns the software.

An OEM employee hinted at the fact that software products are truly black boxes, another one admitted that the company loses its edge by not developing its own software, but yet another one mentioned that they have a strategy to focus more effort on in-house software development.

It is therefore not possible to know who exactly will "own" the software, but it is clear that whoever owns it will gain tremendous leverage.

V. Value Migration in the Design Chain

V.A. Value Creation and Value Migration

V.A.1. Mountain range analysis

An overview of the industry shows that there are few pure players. For example, Bosch who is the number one automotive supplier in the World is diversified not only within the automotive industry, but also horizontally across industries. Its business is divided in *automotive technology*, *industrial technology* and *consumer goods and building technology*. The automotive sector is however its biggest business and represents about 70% of its sales. Within the automotive sector, it supplies a variety of product systems such as Gasoline and Diesel Systems, Chassis Systems Brakes and Controls (that include both the mechanical parts as well as the electronics and software), Energy and Body Systems (there again, it comprises mechanical sub-systems as well as electrical and electronic), Car Multimedia, Automotive Electronics (sensors and control devices), Steering Systems, and Aftermarket.

Very quickly, the work required to perform a mountain-range or layer analysis of the automotive industry as detailed as what Baldwin & Clark did for the PC industry becomes extremely tedious. It is not possible to separate distinctly electronic from mechanical from software components like what they did with the PC industry where the architecture is modular to such an extent that companies are very much focused on one or few sub-sectors only (see Figure 7).

A similar study was however attempted. All North American public companies filing in the US whose main products are related to the automotive industry were researched. For each of them, the following data was gathered:

- Lowest stock price during fiscal year
- Highest stock price during fiscal year
- Number of shares outstanding at the end of the fiscal year

An estimation of the market value of the company was then calculated by multiplying the number of shares outstanding with the average of the lowest and highest stock price. The market value for each year was then adjusted for inflation and normalized in 2004 US dollars.

Each company was then classified according to the following segment decomposition:

- Car manufacturer: these are OEMs
- Electrical/Electronics: companies whose main business is to provide electrical and electronics systems
- Exterior: companies that supply body components (e.g. sheet metal, bumpers, windshield, etc.)
- Interior: companies that supply all systems that are located within the cabin (e.g. door trim, seats, dashboards, carpeting, etc.)
- Chassis/Powertrain: companies that supply what is found under the hood and chassis (e.g. engine mounts, brake systems, suspension modules, etc.)
- Multiple: those companies that are involved in more than one of those previous segments and where one segment is not particularly dominant
- Aftermarket E/E: companies whose main business is in aftermarket products that are electronics and electrical related
- Aftermarket non E/E: companies whose main business is in aftermarket products that are not electronics and electrical related.

The results were tabulated and are summarized in Figure 21. Note that the vertical scale that represents market value is logarithmic because there are several orders of magnitude that separate the various segments. This plot does not depict such a radical evolution as what the PC industry has experienced. For one, the automotive industry is much more mature than the PC industry, but more importantly, it has a much lower clockspeed; thus, any change that would occur in the industry is likely to take much longer.

We can however observe some interesting trends. Particularly in the past 20 years or so, the aggregate value of companies within the "interior" segment has considerably risen by more than an order of magnitude (all values are in 2004 dollars) while the value within OEMs has only slightly risen and that of Powertrain/Chassis components suppliers even less so. What this may indicate is that the value in a vehicle increasingly lies at the interface with the driver and passenger. In other words, customers recognize value in what they see and interact with. Engines and suspensions certainly have a tremendous value in the eyes of the customer, but the relative value of interior components seems to have risen.

More importantly, as Figure 22 clearly shows, the aggregate market value of all North American suppliers has steadily increased over the past fifty years (all values in 2004 dollars). In contrast, the market value of North American OEMs has varied greatly and its 2004 level was about the same as in 1960 while 1965 and 1999 were peak years while the lowest value was attained in 1981. This is approximately the time when OEMs started outsourcing their components more aggressively. Suppliers thus grew in size and even hired employees from OEMs.[10] This represented the human value aspect of the whole value migration. The stability of the suppliers' growth vs. the volatility on the OEMs side is probably due to the fact that the suppliers' market value is aggregated among many players while there are only a handful of OEMs.

The distribution of the total industry market value between OEMs and suppliers (Figure 23) confirms the trend that suppliers have captured an increased share of the total market value.

It is important to note that such companies as Freescale (biggest chip manufacturer for the automotive industry), Panasonic, Bose, and many other companies who originate from other industries but supply more and more components to automotive manufacturers, are not included in the market value analysis that was performed, because their primary business is not directly related to the automotive industry. However, they do capture some value in the automotive industry but it is difficult to assess how much.

Figure 24 is an attempt to draw what the distribution of the market value in the automotive industry would look like if those companies with roots in other industries were taken into account. While it is a speculative chart, it suggests that OEMs' share of the total industry market value is shrinking more than Figure 23 shows.

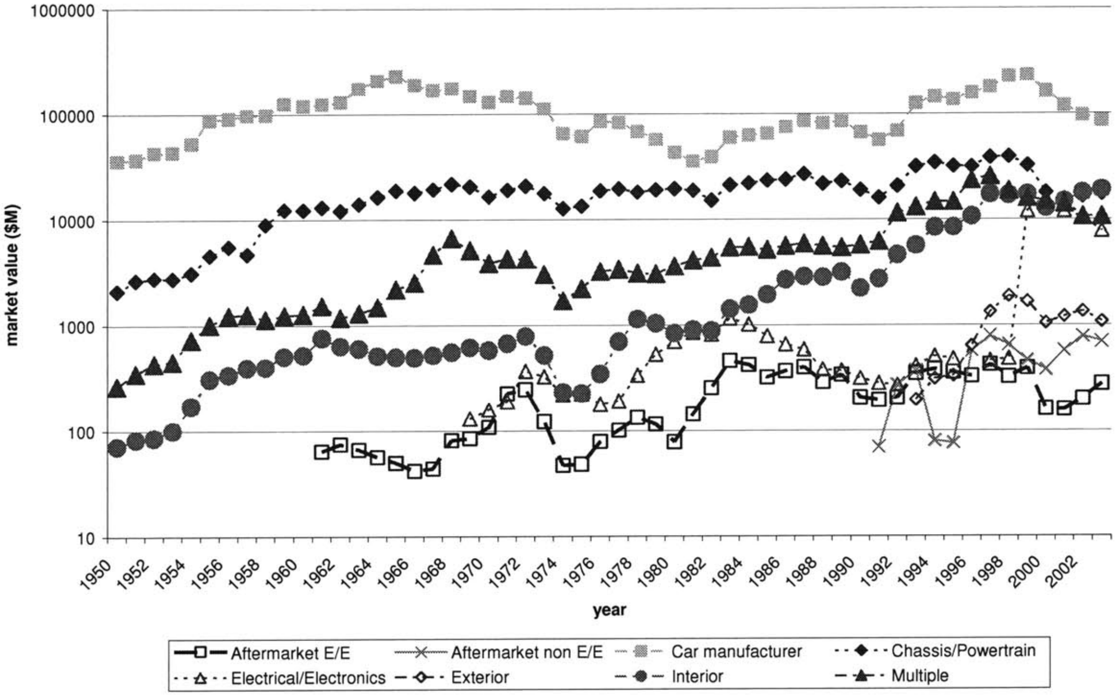


Figure 21. Automotive industry market value decomposed by segments

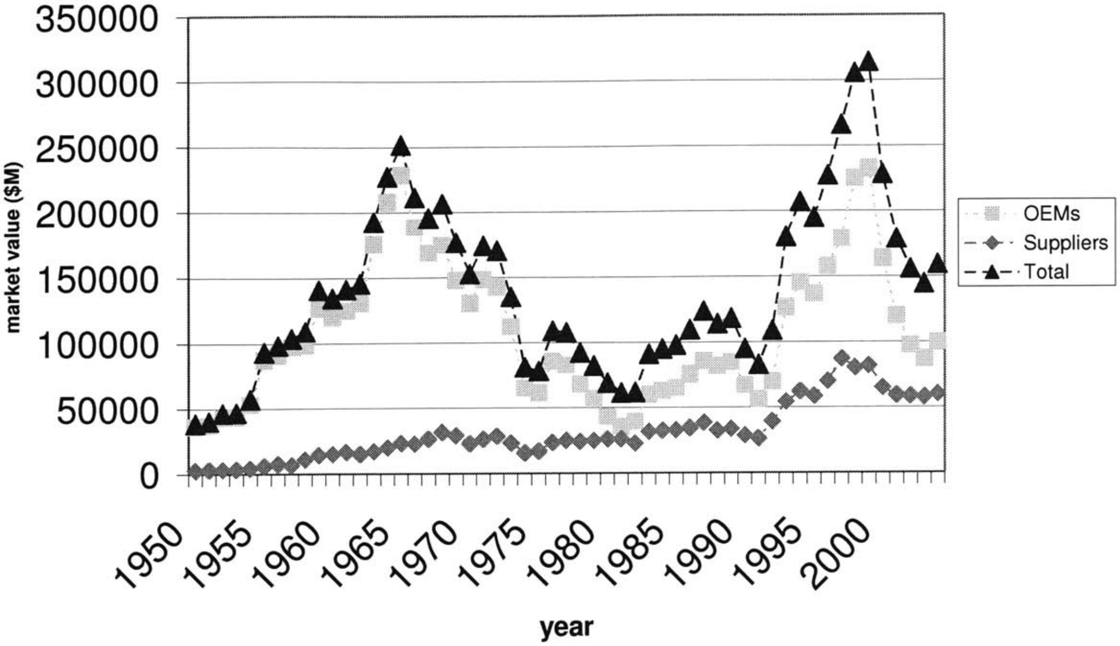


Figure 22. Market value of OEMs vs. suppliers

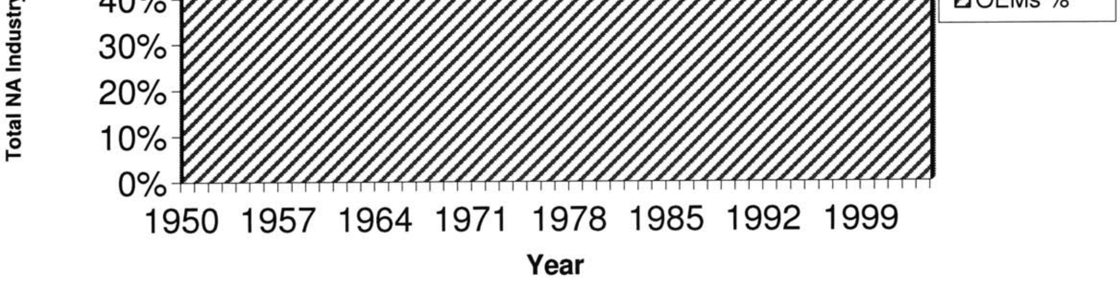


Figure 23. Distribution of market value between OEMs and suppliers

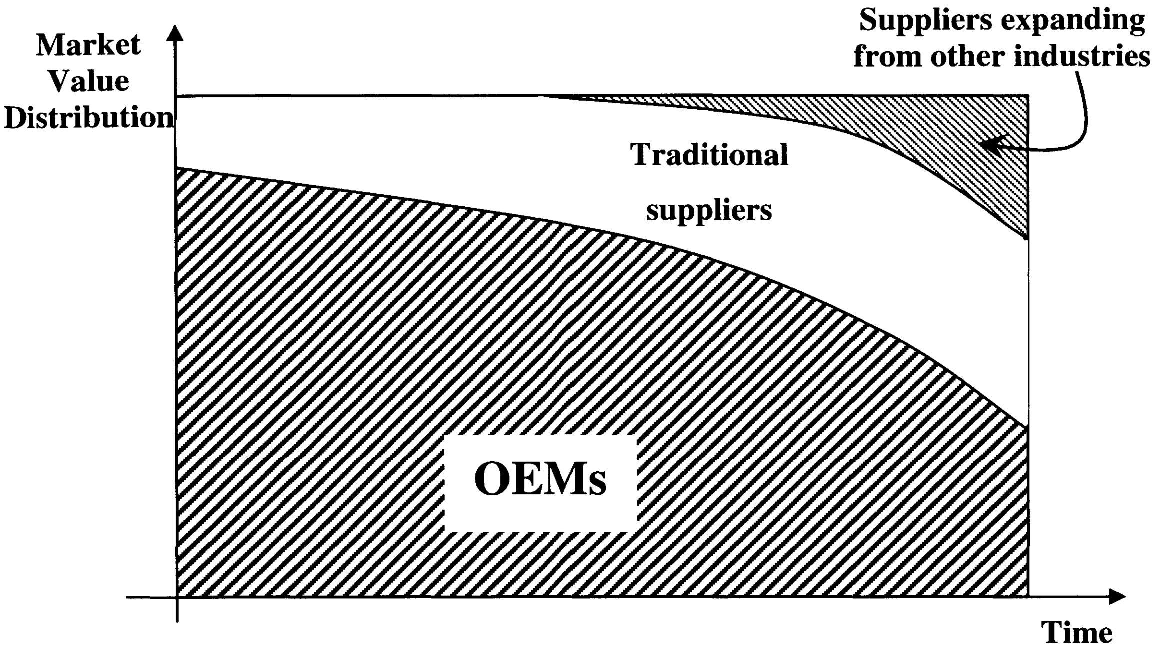


Figure 24. Proposed model of the evolution trend of the market value distribution in the automotive industry

V.A.2. Where Do Innovations Come From?

Active electromagnetic suspensions: MIT professor Amar Bose, founder of the Bose corporation, famous for its speakers business, has developed a new shock absorber based on electromagnetic technology. Replacing springs and dampers is a device composed of linear electromagnetic motors, coupled with high-power amplifiers and governed by complex control algorithms. Dr. Bose claims this new system is a remarkable step forward in ride and handling but it is a pricey solution. While the future of this new system is uncertain, it is interesting to note that a company that is not directly related to the automotive business is trying to enter it by offering a new solution to a typical automotive issue (that of rough ride and unresponsive handling, and more specifically the compromise between the two) with a new technology.

Actively tuned suspension: there have been several types of innovations related to car suspensions. Citroën has been for a long time a leader in hydraulic suspension. Lotus had developed fully active suspensions for its Formula One cars. Nissan introduced a fully active suspension system for the Japanese market only in 1990. Delphi has recently developed a semi-active suspension that uses magneto-rheological fluid, a fluid whose viscous properties can be controlled via a magnetic field. Kinetic, an Australian company has developed other types of semi-active suspensions. It was acquired by Tenneco[53].

Airbag: Allen Breed is credited with the invention of airbag system in 1968 (essentially the sensing system) although earlier patents had been granted to Walter Linderer for more rudimentary designs[54]. Airbag technology was only implemented by GM and Ford after federal government required passive restraints in automobiles in 1990[55] and after timid trials in the 1970s and 1980s.

Anti-Lock Braking System (ABS): the first patent for a brake force controller was filled by Karl Wessel in 1928 but was never transferred into a product. Then Robert Bosch and Fritz Osthaus performed some ground work but it is not until 1970 that Heinz Leiber, who worked at Teldix then at Daimler-Benz, brought ABS to a prototype. According to Heinz Leiber, ABS was "a pioneer in digital electronics".[56]

Automatic transmission: Ralph Teetor, inventor of the cruise control, is also credited for inventing the automatic transmission in 1921, but there was little interest at the time[57]. The first automatic transmission to make its way into production was the Hydra-Matic developed by General Motors' Oldsmobile division in 1940. It used a fluid coupling to transmit torque. Eventually, General Motors, Chrysler and Borg Warner (for Ford) developed further automatic transmission designs, in particular basic fluid coupling led to the more complex hydraulic torque converter in the 1950s.[58]

Catalytic converter: Trinity College is credited with the invention of the catalytic converter[59]. However, John Mooney and Carl Keith, from the Engelhard corporation, are said to be the principal inventors of the three-way catalytic (TWC) converter and commercialized it in 1976[60]. Robert Stempel, from General Motors, ensured it was implemented on vehicles[61].

Continuously variable transmission: the first sketches of a continuously variable transmission (CVT) were actually drawn in 1490 by Leonardo da Vinci. The first patent for a toroidal CVT was awarded in 1886, Adiel Dodge received one in 1935. But it is Dutchman Hub van Doorne, cofounder of Van Doorne's Automobiel Fabriek (DAF) who introduced it in his cars in the late 1950s. Subaru starting offering CVT in its vehicles in the 1980s, followed by Nissan in the 1990s, Honda, Toyota, Ford, Audi have since been offering optional CVTs in some of their vehicles.

Cruise control: The first speed control used a centrifugal governor and was invented by James Watts and Matthew Boulton in 1788 for steam engines on locomotives. It was first used on automobiles in 1910. The modern cruise control was invented by Ralph Teetor. In 1945, he received a patent for what he called at the time a "speedostat" that uses a solenoid to vary throttle position as needed [62]. It is not until 1958 that the cruise control was first offered in a Chrysler car.

Electronic Engine Control: Early analogue engine control systems were developed and implemented by GM in partnership with Motorola (now Freescale) in the 1970s[63]. The modern digital electronic engine control was originally developed for Formula One racing in 1987 before it logically and quickly made its way into production cars [64].

Fuel injection: Fuel injection has been used on Diesel engines for over 100 years, but they have been mechanically controlled, leading to less than optimal mixture, efficiency and pollution control. Bosch developed an injection system for gasoline engines in 1955. Bendix worked on an electronic fuel injection system but the lack of solid state sensors or availability of transistors led Bendix to abandon its development and sold its patents to Bosch. With stricter exhaust emissions laws, Bosch developed the first electronic fuel injection (EFI) system which was used on a Volkswagen in 1967. Bosch continued the development of the EFI which evolved from an open loop to a closed loop with feedback from oxygen sensors controlled through the ECU.[65]

Head-up display: HUD was developed for military fighter jets pilots who encountered information overload issues and for whom bringing the eyes down to look at controls inside the cockpit could potentially be a fatal distraction. It then made its way into commercial aircrafts and is now offered on passenger cars.[66]

Lane departure (road recognition): Various universities (Carnegie Mellon, University of Michigan, University of California, Ohio State University among others) are working at developing warning systems that would alert a driver whose car drifts away from its lane. Valeo and Iteris have developed a lane departure warning (LDW) system that is now offered in Nissan vehicles.

OnStar: Although an IBM ExtremeBlue team (named Blue Octane) is credited with originally developing the concept of OnStar[67], it seems that GM actually developed it. OnStar originated in 1991 or 1992 as Project Beacon. The idea was to provide services and collect revenues after a car was sold. There have been several iterations since its beginning about 15 years ago[68].

Power steering: Francis W. Davis and George Jessup invented power steering in the 1920s in Waltham, Massachusetts. It was put into production by Chrysler in 1951 and marketed it as the "Hydraguide"[69].

Satellite navigation system: researchers at the Massachusetts Institute of Technology realized, soon after the Russians launched their satellite Sputnik in 1957, that the relative position between the satellite and the receptor of its radio signal could be estimated thanks to the Doppler effect of that signal. A positioning system based on satellites signals was implemented and used by the US Navy. The Department of Defense then launched the Global Positioning System (GPS). By the mid 1990s, there were 24 operational GPS satellites. In 1997, the DoD made available to civilians the possibility to use GPS.[70]

Self-starter: Invented by Clyde Coleman in 1899 and received a patent in 1903. But his design was impractical. Delco purchased his patent, and General Motors acquired Delco. GM modified Coleman's design to make it practical and put it into production for the first time on Cadillac cars in 1912.[71]

Variable timing camshaft: General Motors was first to experiment with controlling the intake valves relative to engine speed in order to reduce emissions. They encountered problems however and did not pursue the research. In the 1970s, Fiat developed a functional system whereby lobes – whose shape dictates the timing of the intake and outlet valve openings – were cut in a 3 dimensional way and a hydraulic system moves the camshaft linearly sideways so that at a specific speed corresponds the lobe profile for optimal valve timing. Many other car manufacturers followed suite (such as Honda and BMW in particular) and developed their own specific variable valve timing designs.[72]

It is interesting to note from Table 1 that most innovations originated from independent inventors, small enterprises or suppliers. Those that came from OEMs tend to be related to the internal mechanisms of engines.

Innovation	Inventor	Year invented	Implementer	Car manufacturer	Year implemented
Self-starter	Clyde Coleman	1899	Delco	GM	1912
Automatic transmission	Teetor	1921	GM	GM	1940
CVT	Da Vinci	1490	Van Doorne	DAF	1950
	Dodge	1935			
EFI	Bosch	1955	Bosch	Volkswagen	1967
ABS	Wessel	1928	Teldix	Daimler Benz	1970
Variable timing camshaft	GM		Fiat	Fiat	1970
Airbag	Allen Breed	1968	Allen Breed	GM & Ford	1970s
Electronic Engine Control	GM & Motorola	1970s	GM & Motorola	GM	1970s
Catalytic converter	John Mooney and Carl Keith (Engelhard)	early 70s	Engelhard	GM	1976
Actively tuned suspensions	Lotus, Citroen		Nissan	Nissan	1990
OnStar	GM	early 1990s	GM	GM	early 1990s
HUD	Military aircrafts			GM	
Satellite navigation system	MIT, DoD	1957			1997
Lane Departure	Various universities		Valeo, Iteris	Nissan	2006
Active electromagnetic suspensions	Bose	2005	Not yet implemented		

Table 1. Timeline summary of select innovations in the automotive industry

V.A.3. Discussion

Clayton Christensen wrote the following quote in an edition of the Harvard Business Review:

> *"Products are most profitable when they are not "good enough" to satisfy customers' needs. This is because to make them performance competitive, engineers must use proprietary, dependent architectures. Use of such architectures makes product differentiation straightforward, because each company pieces its parts together in a unique way.*
>
> *Once a product's performance is good enough, companies must change the way that they compete. The innovations for which customers will pay premium prices become speed to market and the ability responsively and conveniently to give customers exactly what they need, when they need it. To compete in this way, companies are forced to employ modular architectures for products. Modularity causes the products to become undifferentiable and commoditized.*
>
> *Attractive profits don't evaporate, however...*
>
> *They move elsewhere in the value chain, often to subsystems from which the modular product is assembled. This is because it is improvements in the subsystems, rather than the modular product's architecture, that drives the assembler's ability to move upmarket towards more attractive profit margins. Hence, the subsystems become decommoditized and attractively profitable."*[73]

Arguably, one could say cars have been "good enough" for a while and OEMs have modularized the automobile architecture, have been competing increasingly on time-to-market and have relied on outsourced modules to differentiate their products. Hence, the value has migrated over to the suppliers, and this is what we observe indeed. This also appears to be in line with Baldwin & Clark.

There is however a difference. Christensen ascertains that the modularization of an architecture "causes the product to become undifferentiable and commoditized". Baldwin and Clark on the other hand do not seem to believe that the consequences of modularity are so clearly predetermined. They argue that a modular architecture "is a financial force that can change the structure of an industry", a "*Darwinian world [...] of growth, innovation and opportunity [...] that can also fall into periods of extreme value destruction*"[24]. By loosening the boundaries in the design chain structure, the modularization of an architecture destroys a relatively stable state, thus engendering a dynamic reshuffling within that design chain. However, the resulting effects are not necessarily known in advance.

Open standards – and in particular AUTOSAR – are creating this destabilizing effect. They create a set of common design rules and foster encapsulation of software components – two of the three necessities for a modular design according to Baldwin & Clark. This in turn generates modular boundaries in the design chain and sets in motion a new dynamic in the industry[i]. The third necessary component of a modular design however - system integration and testing - cannot be dictated by such a standard. And this is the element that will dictate whether a company can take advantage of this new dynamic or will suffer the consequences of value migration.

As was mentioned earlier, Baldwin & Clark identified three different types of modularity: modularity-in-use, modularity-in-production and modularity-in-design. It would appear that Christensen may have considered the first two, but perhaps not the last – and arguably most important – one. Indeed, modularity in design implies that the system value increases as its modules gain value. In other words, the value does not migrate outside, but remains within the system. Baldwin & Clark's concept of modularity in design implies that innovations can be undertaken and implemented on modules without disrupting the whole system. Because more modules mean more opportunity for

---

[i] As David Sharman mentions, modularization is not standardization and vice versa. However, the two are tightly connected although the causal relationship is not trivial. Indeed, the need for standard interfaces leads to the standardization of those interfaces, similarly standardization encourages a decomposition into modules.

improvements, modularity is a powerful source of numerous improvements that benefit the whole system. On the other hand, Henderson and Clark differentiate between incremental (component) improvements and radical innovations, as well as modular and architectural innovations[74]. Modular innovations may correspond to new technologies that improve a component or module, while architectural innovations are improvements in the way those components, modules or sub-systems interact with each other. In other words, modular innovations affect modules and architectural innovations affect their interfaces and integration as a whole system. It is clear that both are not mutually exclusive. In fact, what Baldwin & Clark seem to suggest is that modularity in design is made possible by the architectural design. This suggests therefore that architecture not only determines the emergence – i.e. the appearance (desired or undesired) of function whose functionality is greater than the sum of its parts[75] – but it also dictates the aptitude of a design to capture value from modules and to enable those modules to develop value independently.

When focusing on modularity-in-use, one observes that the customer can freely choose modules independently, mix and match them and come up with a customized system. The customer sees the value in the modules themselves, and takes the standard interfaces between modules for granted.

With the modularity-in-production lens, it is easy to see how the value may migrate. If modules suppliers use proprietary designs and keep the module integral, they can retain value, while the activity of putting together the modules of the customer-assembler may have little value.

The concept of modularity-in-design is defined within the design chain architecture. By creating clear interfaces between modules, transactions are reduced, thereby reducing cost and complexity, but more importantly, the constraints within the design space are minimized. As a result, each module can bring additional value to the system. This notion is a reflection of the paradox of power: the more power one shares, the more power one gets. By weaning out suppliers, a customer gives them more freedom to develop better products.

But saying that modules that inherently gain value can bring that value to the system is a different proposition than saying that those modules actually do bring the value to the system. Furthermore, even if the system does indeed gain in value as a result, what assures that the system assemblers profit from it?

In "The Power of Product Platforms", Meyer & Lehnerd write of a company that had standardized key subsystems and components that "*the company was finally able to make money in all market segments. Rich in features and achieving economies through standardization in mechanical function and design, the manufacturer had created a far more profitable and exciting line of products*". Throughout their book, they give examples of how companies create value by creating platforms at the basis of which is the standardization of interfaces. One of these examples is that of Black & Decker: by using common interfaces, modularization of functions was achieved (such as power with every motor sharing the same interfaces and being a scaled version of a single design) many parts could be re-used, leading to significant price decrease (by a factor of 4) and functionality could be tailored to better target customers.[26]

The examples from Meyer & Lehnerd help answer the questions raised above. The key is decomposition. A good decomposition is performed by embracing various domains such as marketing, engineering, purchasing, etc. The decomposition task should not be performed with respect to a single one of them. This is unfortunately too often the case as we will see in V.B. It is through a decomposition based on a systemic strategy that a company can ensure that it captures the value created by modularization. If a company's modularization strategy is based on only one dimension – say economies of scale only – then a bigger picture based on system design is lost. A big picture design approach includes a strategy for integration of the modules.

Most importantly, in all the successful examples that Meyer & Lehnerd gave, integration is an upstream process, it drives the decomposition into modules based on the company strategy. Value resides in the architecting of the system, that is, in thinking up how every sub-systems will fit together to create functionalities. When integration becomes a reactive process – because it was not part of the strategy – it is very difficult to capture the value.

This shows the importance of Baldwin & Clark's notion of modularity-in-design. The comprehension of this notion is an essential prerequisite to understanding how to capture value created by modularization.

As it stands in the automotive industry, OEMs seem to have a sense of the dynamic that is currently changing the industry and have incorporated into their overall strategy a goal to bring more control of software in-house. This is a step in the right direction. What seems to lack however is the realization that, in order to "control the controls", a major knowledge-based shift has to occur which requires investment in the development of new, less mechanic-centric processes, combined with a strategy to bring in the knowledge necessary to implement this change (either from employees from other industries, new college recruits, or from partnership with universities, academic and professional organizations and companies from other industries). Such a change is necessary to ensure that a proper integration strategy is in place.

Figure 25 is an overview of the dynamics that occur between modularization, knowledge retention and value migration. There are six loops in this diagram.

1. <u>M&A – more internal R&D:</u> This is a reinforcing loop where increasing complexity of a product means the required knowledge to remain in the business also increases, as a result, a company will merge with or acquire another company and the internal R&D will therefore increase, this in turn leads to more complex products.
2. <u>Value through R&D:</u> with increased R&D, a company develops more internal exclusive knowledge and is able to deliver more exclusive functionalities. This translates into higher company value and better financial performance. As a result, more capital is invested into R&D. This is a reinforcing loop.
3. <u>Modularization / Value migration balance:</u> this is a balancing loop. With increased complexity, the need for modularization is higher which eventually leads to more modularity in the product architecture. As we have seen, more

modularity leads to increased outsourcing as design of sub-systems may be more easily separated. Inevitably, outsourcing means that customers and suppliers have to communicate, need to understand each other, therefore they share knowledge. Their share of exclusive knowledge is then diminished (they focus on what they consider their core competencies). The potential for truly exclusive functionalities is lower, which in turn diminishes the value of the company and its performance which means the company's R&D is likely to be affected negatively and less complex alternatives are then sought.

4. M&A – less outsourcing: If a company chooses to acquire or merge with another company, its outsourcing needs will be reduced. As we have seen in the previous loop, this leads to less knowledge vanishing (more knowledge retention), more exclusive knowledge which eventually means better performance and more investment in R&D. Complexity is increased and more knowledge is required, leading to more need for M&A of internal investment in R&D. This is a reinforcing loop.
5. Modularity self reinforcement: as we have seen, with increased modularity comes more outsourcing and more knowledge vanishing. As a result, more knowledge is out in the open. This leads to more standardization, which leads to a more modularizable architecture and eventually to more modularity. This is a reinforcing loop.
6. Value through modularity: This is a reinforcing loop that is based on Baldwin & Clark's theory. More modularity means that there are more subsystems per product and therefore more design options (see III.A). This yields a higher system-product value, which brings more value to the producer. With more value and better performance, R&D is increased, leading to more complex products and eventually additional need for modularization. This applies mainly to OEMs but more generally to a company that has the ability to modularize its product architecture.

From this diagram, note that "Product Modularity" is involved in three loops – two reinforcing, and one balancing. The "Modularity self reinforcement" loop shows that modularity tends to call for modularity, but the weakest link is perhaps the standardization because the extent to which it may occur and the delays that may be involved with it are quite variable. The two most important loops are "Modularization / value migration balance" and "value through modularity". In the former, product modularity leads to decreased value to the company (OEMs or any company that modularizes its product) because of the knowledge that flees away from the company, in the latter, it brings value to the company because of the increased number of value options. An approach whereby keeping the architecture of the product as integral as possible may help in preventing that knowledge flees away, but as a result, product complexity may suffer and value creation is limited because of the impossibility to profit from various design options. A proper management of the modularization of a product architecture is necessary. This requires an understanding of what the exclusive functionalities that actually bring value are, this should drive the decision of which knowledge is important to keep in-house, and this in turn ought to drive what to outsource and how. More importantly, the "product design value" loop indicates that there is a necessity to have the knowledge in-house to capture value from various design options that modularity provides, this in fact is the essence of system engineering. The "modularization / value migration balance" loop is a representation of the exercise of adequately decomposing the architecture of a product and doing the make-buy decision so that knowledge that will eventually bring value to the company is not lost to suppliers or competitors. The "value through modularity" loop illustrates the fact that the architecting of the product should be such that the value provided by the modules can effectively be created and captured.

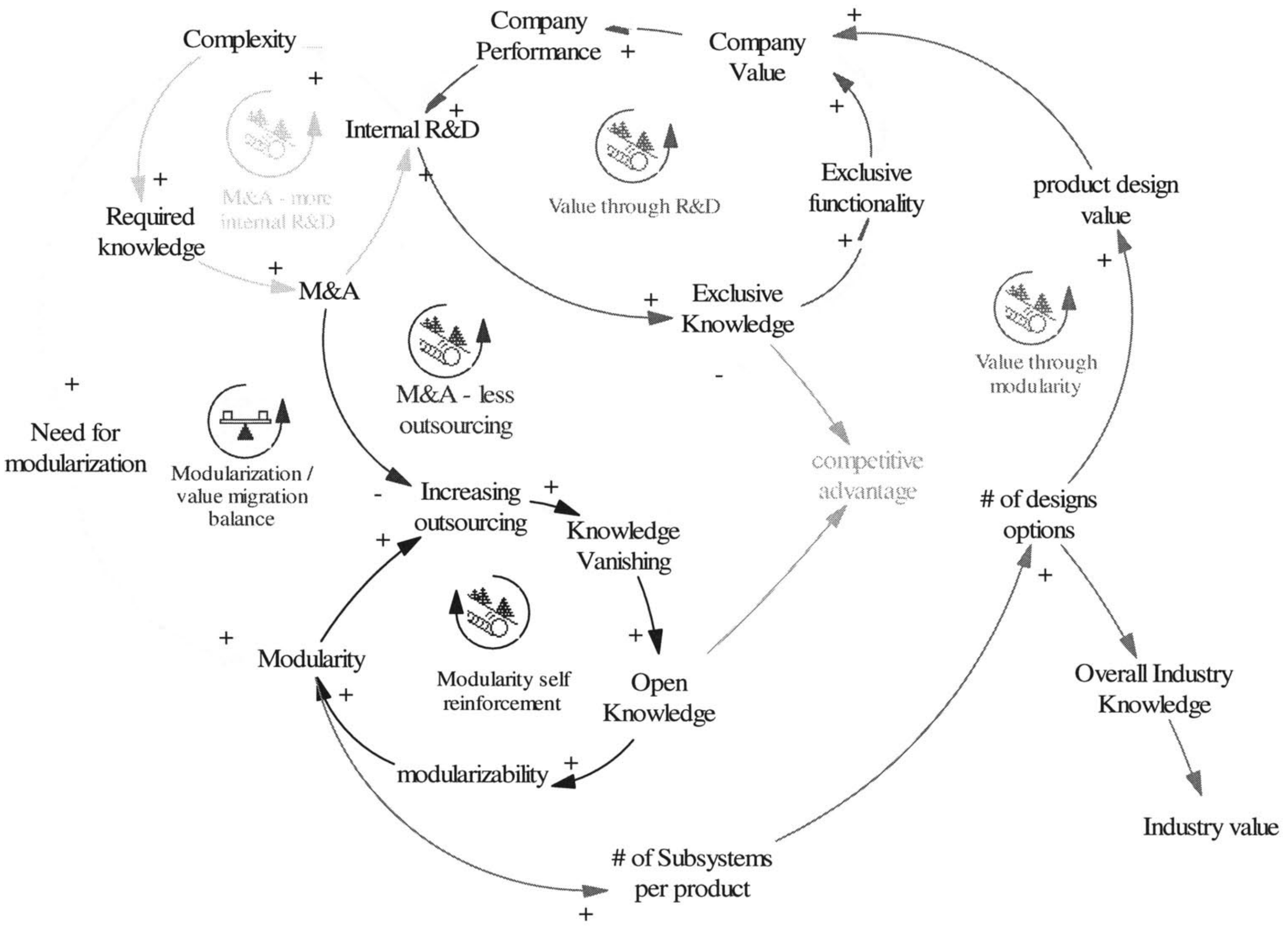


Figure 25. System Dynamic approach of the issue of modularization and value migration

V.B. The Purchasing-Centric Strategy Trap

"The architect is not a generalist, but a specialist in simplifying complexity, resolving ambiguity and focusing creativity"[76]

- Professor Edward Crawley

As we have seen in II.B.2, decisions with regards to modularization of the architecture should be made with consideration for the overall company strategy as well as a balance between the cost of modularization vs. the value that it may create and that may be captured. In order for this to happen, it is very important that several departments work in concert, and in particular purchasing and engineering. As Timothy Thomas points out, "*In general, if purchasing does not work together closely with engineering (and vice versa) each activity can effectively undermine the other*"[77].

It seems however that there is a lack of balance between the role that purchasing plays vs. engineering's role on affecting the modularization of the architecture in many automotive companies. Another OEM employee notes that by seeking buying power and low prices from suppliers, this OEM's purchasing department wants to buy SKUs in large quantities and thus makes decisions on where there should be economies of scale and scope. He notes that while the engineering's impact on such decisions is much less, the impact on engineering is great. Indeed, engineering now has to deal with the decisions made by purchasing that often represent additional design constraints.

There is an amalgam between the search for reduction of complexity and the consideration of cost. While those matters are certainly related, the driving force ought to be complexity as cost usually depends on it. Unfortunately, too often, considerations of cost are put forth and the management of complexity is left for engineers; but by then those engineers have their hands tied. Such a dynamic may result in making poor product-system decomposition and may lead to less-than-optimal outsourcing choices which leads to the evasion of competency.

If decisions are made by looking only through the purchasing lens, then OEMs may lose strategic competencies and value migration in the industry shifts toward those suppliers who retain their core competency and own the value-rich systems.

In their paper about the Make-Buy Decision process, Professors Fine & Whitney explain that a Japanese OEM retains knowledge of half-shafts design and manufacturing even as they outsource it. They outsource because they feel that they may not have the capacity for one reason or another to make all the half-shafts they need. However, because they feel it is an important element of the system design and a safety critical component, they do not want to be dependent on their suppliers for design and manufacturing knowledge. They therefore keep the process knowledge in-house by maintaining a minimum design, development and production process workforce who runs a minimal volume production[9].

In the case of software products, there is not really such as thing as outsourcing manufacturing. There are however components that are considered commodities – such as databases – and dependency on capacity could be in the form of licensing or customer support for these commodities. Dependency on knowledge on the other hand may exist for integrated components and specifically targeted functionalities.

Software development is inherently uncertain, flexible, adaptive and iterative, so that specifications and implementation are tightly coupled. Consequently, if requirement analysis and software development are decoupled, learning opportunities are lost and knowledge inevitably leaks out. This phenomenon is amplified by the fact that software is easy to replicate and adapt.[78]

Making the right choices is thus important from a knowledge retention perspective, but this necessitates a certain understanding of what ought to be outsourced, what should not be, and how it all fits together. If this task is left to the purchasing department alone, it may be quite difficult to retain knowledge.

According to professor Thomas Keim, AUTOSAR was in fact an effort initiated not by engineering teams of the various founding companies, but by their purchasing departments who seek economies of scale.[79] This consortium seems well thought out from an architectural point of view, and therefore provides a good basis for the development of a standard. If a company takes a holistic approach about it, such a standard could potentially help reduce complexity, cost and bring value in the long run. But if a company enters into this consortium with motivations driven solely by purchasing, it may miss a bigger picture and end up making poor strategic decisions.

Perhaps the aspect that may make AUTOSAR difficult to "take off" is the short-sightedness of some executives, particularly on the supplier side. If a supplier wants to start making AUTOSAR-compliant products, it may have to make changes to its architecture, designs, outsourcing, etc. it would require a significant amount of investment. Standard interfaces would decrease switching cost for customers; on one hand, this means a supplier may sell the same product to several OEMs, on the other, it also means that OEMs may easily switch to other suppliers. As a result, the biggest suppliers, those that may have the means to invest into making their products AUTOSAR compliant, are the ones who may be reluctant to do so because they already have a big customer base, while smaller companies looking to extend their market share may want to ensure AUTOSAR compliance, but may not have the means to do so.

In the long run however, such an open standard would help a company focus on the value-rich features, reduce complexity and cost. But in order not to fall in a trap where cost is the driver and the focus on value-rich features is neglected, system architects ought to be involved. It is the architect who should drive the modularization of the products through a systemic strategy.

Baldwin & Clark explain that architectures are what channels knowledge so as to turn it into value, the link between knowledge and the economy.[80] Architecture itself is not the value, it is not what is sold to the customer, but it is the enabler of value creating and capturing that leads to designs – the abstraction of value – that is materialized by a product. The product is what is sold, it is what holds the value that the customer is willing to pay for, but the architecture is what infuses value into the product.

An organization therefore ought to focus on the processes that lead to the making of a product, not on the product itself. Focusing on the product may lead to failure to infuse value but also failure to control the value infusion, thus allowing value to migrate to those organizations that do play the role of architects.

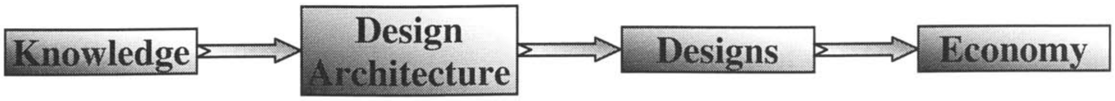


Figure 26. Architecture is the link between knowledge and the economy (from Baldwin & Clark[80])

V.C. The Disruptive Technology Lens

In “The Innovator’s Dilemma”, Clayton Christensen defines a “disruptive technology” as a technology that is simple but challenges existing business models, usually one that enables applications to move down market to new uses. As such, it is disruptive to established competitors of that use.[81]

He also distinguishes between “traditional” and “ancillary” performances and argues that a disruptive technology has worse traditional performance but better ancillary performance than the displaced technology.[i]

In the case of automotive electronics, there are several traditional performances that may be identified, sporty behavior performance (power, speed, handling), comfort (ride, sound insulation), maintenance (maintenance cost, functional failures, repair needed). Most of those relate to the car as a mechanical machine. Ancillary performance on the other hand would tend to be more related to the new paradigm of the car as a multimedia center on wheels (integration of cell phone, navigation system, access to internet, control through voice recognition, etc.).

Most customers buying cars nowadays still regard mechanical performance as a strong factor that influence their decision. And in fact, electronics helps in this domain by providing better control (e.g. active engine mounts, engine management, etc.). For automotive electronics to be considered a disruptive technology according to Christensen, that “traditional” performance would have to be lower while the identified “ancillary” performance is better. Automotive electronics technologies are therefore not disruptive in the Christensen sense of the term.

That type of disruption may however come from the push for alternative power sources. According to a Hart Research report, the top priority for US energy policy is to reduce dependence on foreign oil, which, with 43% of the votes, is considerably ahead of

---

[i] For a more in-depth description of Christensen’s model of disruption, the reader is encouraged to refer to “The Innovator’s Dilemma”

the second most important priority of improving fuel efficiency of vehicles (20%), the third priority, with 19% is to reduce pollution and harmful emissions, and the fourth one, with 15% is to keep fuel costs low. The report also shows that the technologies developed by OEMs follow approximately the same ranking in terms of importance. Similarly, consumers' focus is on gas mileage (36%) as much as safety features (36%) followed by alternatives to gas engines (32%); pollution and reliability come next (21% and 16% resp.) while performance and styling seem to have much lower influences (6% and 4% resp.).[82]

It appears therefore that the interest for alternative automotive power sources is growing among various stakeholders. While alternative power sources may suffer in terms of traditional mechanical performance, they surpass the current design of vehicles in ancillary performances – such as fuel consumption, reduction of pollution, decrease of dependency on oil – for customers but also importantly regulatory organizations. As such, they satisfy Christensen's model of disruption.

The role of automotive electronics in enabling a shift toward alternative power sources for next generation vehicles is undeniable. Better fuel mileage for hybrid vehicles for example may be achieved through a better management of the charge/discharge of electric batteries thanks to an elaborate control algorithm (see page 28); communication protocols such as Byteflight and FlexRay provide the necessary tools for the development of technologies required for communicating between safety-critical components in a more electric vehicle (see page 31).

Therefore, although automotive electronics in itself is not a disruptive "technology" according to Christensen's model, it may as well be considered as such because it enables the implementation of one.

But what type of disruption is the automotive industry likely to sustain? And who/what is at risk of being disrupted? The GM Hy-wire (see page 77), although not a mature design for mass market, provides an insight of how vehicles' architecture may evolve; thanks to electronics, functionalities will be clearly more separated than they are now. With such an architecture, and with increasingly standardized and commoditized hardware, one may envision future potential new entrants in niche markets at the low end of current markets, aimed at specific applications where the traditional performances of current vehicles (i.e. speed, comfort) are not particularly regarded as the most important – e.g. low speed vehicle for city use only, commercial vans that tend to commute in a limited perimeter around cities, etc. Using a Judo Strategy[83] as defined by Professor David Yoffie, a smaller competitor may position itself as a complement to the auto industry vs. a threat or at least avoid to attract the attention of strong establish companies through a proper marketing strategy, it may make its way on the learning curve in small markets before it start moving to more important ones once it has acquired more knowledge. As it does so, it may also leverage established OEMs strength by turning their assets (long-term investments in particular) into liabilities.

There is a major hurdle that is likely to prevent such a scenario to occur. Even as a vehicle becomes more and more modular, the system-product is not simply the result of putting together modules but a system design and integration is necessary. Due to the fact that the basic function of a vehicle is to move people, energy is required and there is a lower bound to complexity (see page 77). The level of that essential complexity is in fact rather high and requires the knowledge of the role of integrator that OEMs have gained over many years and decades.

Moreover, in an industry where profit margins are low, the risks of such a radical move probably outweigh the benefits that may come from it. Suppliers who in fact already tend to play an increased role as systems integrators and who would be the best potential candidate for a move into the OEM realm through forward integration are probably better off pursuing an incremental growth strategy. In other words, it is unlikely that suppliers will attack OEMs directly, but if they play more and more the role of integrators, they may gradually eat an increasing share of the OEMs' value.

The Five Forces framework developed by Porter is useful in evaluating where actual threats may come from.

Five Forces Analysis – OEMs

1. Threats from customers: for OEMs, customers are typically individuals (or car dealers). Although certain customers are loyal to certain brands, an OEM has to ensure differentiation of the brand identity and proper targeting through adequate marketing. That alone is not sufficient as customers are becoming more demanding and expect the same mechanical performance along with better fuel mileage, increased comfort and better reliability, all at the same price. With increased competition from German and Japanese brands in the US and Japanese brands in Europe, the threat from customers is therefore really the threat that the competition does a better job at capturing more market share.

2. Threat of substitutes: It is unlikely that a new means of transportation comes and disrupt the industry. However, following the Christensen's model of disruption, it is not impossible that new technologies be adopted in different value networks. For example, the Smart car in Europe is a new type of vehicle designed for city driving – although it can also be driven on the highway. If customers have enough incentives (through lower price, better fuel mileage, etc.), they may be enticed to change their notion of what a car is or should be.

3. Threat from suppliers: Porter identified that the relative cost of the supplied product provides a certain leverage to the supplier. Clearly, as we have seen in II.B.4.a) the share of electronic components in vehicles is rapidly increasing and is expected to grow even further. However, the cost of hardware or traditional components is still higher today. It would thus appear that the most powerful suppliers are going to be those current suppliers who are integrating electronics and software, not just in their products-systems but more importantly in their organizations.

4. Threat from new entrants: It is improbable that new entrants will disrupt OEMs, be they companies from a different industry, current automotive suppliers integrating forward, or start-ups. Even those powerful suppliers with knowledge of the automotive industry, expertise as mechanical systems as well as electronic and software integrators are unlikely to get into the business of making cars. But

the threat they actually pose to the OEMs is that they will capture the value and negatively affect OEMs' profit margins down the road.

5. Threat from rivals: Clearly, if other OEMs do a better job at developing internal knowledge as not only mechanical systems but also electronics and software systems integrators, they are likely to gain competitive advantage, manage to keep decent profit margins and market share.

Five Forces Analysis – Suppliers

1. Threats from customers: OEMs clearly have a lot of leverage because of their concentration in the industry and the volume they represent. However, the trend has been toward delegating more and more design and integration responsibility to the suppliers. There is therefore little risk of backward integration.
2. Threat of substitutes: As more and more functionality is embedded in electronics and software, suppliers of mechanical systems are at risk to be left with commodity type products if they do not increase their expertise in those domains.
3. Threat from suppliers: as value is migrating down the supply chain, the threat is mainly coming from those suppliers who play the role of integrators.
4. Threat from new entrants: Several companies have already expanded from the high-tech and consumers electronics industries (e.g. Panasonic, Freescale, etc.) to the automotive industry. Freescale – a former division of Motorola that became independent in July 2004 – is the third largest chip maker in the United States and ninth largest in the world, but is also number one in the automotive market where it is experiencing significant growth. Moreover, with the development of open standards, new companies are based around the business of testing mechatronic systems (e.g. ETAS, iSystem, etc.) or debugging software products (e.g. Hitex, etc.).
5. Threat from rivals: with an increasing level of modularity in automotive electronics, the state of the industry has been shaken leading to a reshuffling. Those companies who properly leverage this new trend by acquiring the adequate

knowledge and transfer it into their product designs may capture value. Those who do not may end up in a low-margin commodity business.

Outlook

A look at the aircraft industry (section III.B) showed that after a long initial phase of organic growth and as designs became more and more complex, a series of mergers and acquisitions over the years and decades has taken place. This allowed companies to acquire know-how in certain domains – such as control systems, electronics, electrical network management, material engineering, etc. Had those companies tried and develop these fields of expertise internally, their growth would not have been as rapid and the competition would have had the opportunity to overtake them.

As the expertise requirements in the automotive industry are rapidly evolving, suppliers ought to consider new ways of gaining the knowledge they need to thrive by looking at partnerships, joint ventures and M&A not within the industry but outside its current boundaries, in particular in the domain of software development and integration, and mechatronic testing.

While in the aircraft industry, modularity helps in design reuse from one generation to the other, in the automotive industry, it can also be a tremendous factor in achieving economies of scale. Open standards such as AUTOSAR are thus all the more important in such a high-volume industry.

As the product architecture is evolving, the threat of disruption in the automotive industry does not lie so much in the entry of new players even though there are definitely new players among suppliers. Rather, disruption may stem from transformations in how value is to be captured. The growing amount of electronics and the standardization of interfaces and architecture leads to an increasing level of modularity. This in turn is creating a dynamic in the industry that tends to redistribute value.

Given the current conjuncture, GM and Ford in particular may be reluctant to invest into the capabilities necessary for them to ensure they capture a share of the value created by automotive electronics and thus have their hands tied. As Fernando Cela Diaz explains *"Standard & Poor's estimates that the downgrading of GM and Ford's bond ratings will increase the cost of capital for these two companies and for their close suppliers. The situation is, at best, unlikely to change in the short term, and puts stress on GM and Ford to generate cash and minimize capital expenses; therefore, it is reasonable to assume that both companies will attempt to avoid capital-intensive changes in their supply chains, and favor incremental solutions over radical changes in the core."*[44]. This is all the more hazardous for them as the industry is currently in a shifting phase and potential for value migration is high.

Most OEMs understand the need to invest in alternative powerplant designs. Indeed, many offer more and more vehicles with hybrid powertrains in their line-up. But they ought to understand that a big picture approach is necessary because the integration of new powerplants with other technologies requires mastering the E/E architecture as exemplified by Nissan's hybrid vehicle charge/discharge control system.

VI. Conclusion

Automotive electronics and software are enabling the encapsulation of functionalities represented in Figure 14. Such a separation is leading to new possibilities in architecture development. It increases the modularizability of the car-system, and gives an opportunity to get closer to the lower bound "essential complexity" floor.

As this new type of architecture "wants" to become more modular, open standards are being developed and are likely to have a catalytic effect. Increased modularity often comes with chaotic reactions and reshuffling in the order of the affected industry. In the case of the automotive industry, whoever owns the control will gain a tremendous advantage on cost saving and design opportunities. And controls are held in the software development.

In the design chain, with increasing modularity, the role of system engineering is far from being deprived of any value. Quite to the contrary, it is probably what OEMs ought to focus on if they want to avoid seeing value migrate to their suppliers. The emergence of value is the fruit of architecting. It does not occur simply by putting one module next to another. OEMs should regard the development of new open standards as opportunities to become more aggressive systems architects because system architecting is what determines the value infusion in products. Open standards can also be a way to reduce cost, in particular by creating economies of scale and scope. However, one should keep in mind that reducing cost without creating value is the beginning of a downward spiral.

As software components represent more and more where control lies, OEMs do not have any other choice but to gain expertise in this domain. If they stay only on what has traditionally been "their" side of the architecture – i.e. the mechanical car – their role may end up being marginalized, they will not have any leverage and suppliers will hold them hostage through their electronic control. The black-box nature of software is likely to ensure that software module, particularly as they get more and more encapsulated, remain intrinsically integral for a long time. Outsourcing software (other than the commodity applications) is therefore not a desirable option. OEMs should therefore look at expanding their software expertise and understanding at all ranks of the company and seek to eliminate any dependency to an outside source for software, because it is likely a dependency on knowledge.

Similarly, suppliers may want to do the same and increase their value infusion ability by taking control of the system integration. They should also look outside the traditional mechanical systems box. As Figure 14 shows, systems are now more than just mechanical and hydraulics, they also involve electric, electronics and software engineering. If OEMs do not embrace those, they cannot be complete systems engineers and architects, which leaves the door open to suppliers. However, in order for suppliers to gain the required expertise, they may have to perform strategic mergers & acquisitions, not with peer companies but in the domain of software and mechatronic testing, etc. They should also try and focus on what the end customers see, are aware of and interact with, not necessarily for a direct marketing strategy purposes, but because customers value what they are aware of.

OEMs, suppliers or even new entrants, whoever emerges as a winner in tomorrow's automotive industry affected by new technologies depends on who will take on the role of systems architect because the architecture determines the ability to infuse value in a product, thus to capture value. But in order to be the best system architect possible, a company will have to be an expert in software.

References

---

[1] The Worldwide Market for OEM Automotive Electronics -- 2005 Edition – IMS research

[2] A. Weber “Electronic Road Trip”; 2005 BNP Media

[3] J.D. Power and Associates 2005 U.S. Automotive Emerging Technologies Study^{SM}

[4] F. Andrews “Dell, It Turns Out, Has a Better Idea Than Ford”; The New York Times, January 26, 2000

[5] http://rrrbmn05.waika9.com/cugnot.htm

[6] E. Crawley, SDM System Architecture class notes, 2004

[7] Supplier Business Magazine, Issue 7, 2005, p. 23

[8] D. Whitney “Physical Limits to Modularity”; MIT ESD External Symposium

[9] C. Fine D. Whitney "Is the Make-Buy Decision a Core Competence?" Moreno Muffatto and Kulwant Pawar (eds.), *Logistics in the Information Age*, Servizi Grafici Editoriali, Padova, Italy, 1999, pp. 31-63

[10] Discussion with Professor Whitney, January 2006

[11] N.P. Suh “Axiomatic Design Theory for Systems”; Research in Engineering Design (1998)

12 A. Zakarian “A New Clustering Algorithms and Software for Development of System Architectures for Multiple Product Platforms”; 7th International Dependency Structure Matrix (DSM) Conference

[13] McKinsey & Co, “Managing Innovations on the Road”

[14] Insurance Institute for Highway Safety “Electronic Stability Control Found Effective” October 28th, 2004

[15] Y. Deguchi, K. Kuroda, M. Shouji, T. Kawabe “HEV Charge/Discharge Control System Based on Navigation Information” Convergence Transportation Electronics Association, 2004

[16] http://www.motortrend.com/features/112_0208_trans/index.html

[17] P. Hansen “Software as product” - www.hansenreport.com

[18] McKinsey “HAWK 2015 – Knowledge-based changes in the automotive value chain”

[19] SupplierBusiness; Issue 7 - 2005

---

[20] G. Leen, D. Heffernan "Expanding Automotive Electronic Systems" IEEE Computer; January 2002

[21] Based on McKinsey "Automotive Electronics" report

[22] Discussion with Eddie Khan from the Ford Motor Company; November 2005

[23] Discussion with Mark Schaefer, ONSTAR; December 2005

[24] C. Baldwin, K. Clark, "Modularity in the Design of Complex Engineering Systems" January 2004 (http://www.people.hbs.edu/cbaldwin/DR2/BaldwinClarkCES.pdf)

[25] J. Utterback "Mastering the Dynamics of Innovation" Harvard Business School Press; 1996

[26] M. H. Meyer, A. P. Lehnerd "The Power of Product Platforms – Building Value and Cost Leadership" The Free Press; 1997

[27] R. Liebeck "Design of Subsonic Transports" presentation to the Systems Architecture class, MIT SDM; October 2004

[28] http://en.wikipedia.org/wiki/Airbus_A380

[29] Discussion with Malvern Atherton, Rolls Royce engines; December 2005

[30] http://events.airbus.com "Airbus Sets Up Avionics One-Stop-Shop"; November 14th, 2005

[31] Discussion with Steve Bresnahan, Hamilton Sundstrand; December 2005

[32] http://www.hamiltonsundstrandcorp.com/hsc/details/0,4482,CLI1_DIV22_ETI3212,00.html

[33] http://www.utc.com/index.htm

[34] http://www.ghs.com/

[35] http://www.osek-vdx.org/whats_osekvdx.html

[36] SupplierBusiness; Issue 2 - 2005

[37] American International Automobile Dealers Association "Honda joins Toyota, Nissan to develop car software", October 2005

[38] "AUTOSAR – An industry-wide initiative to manage the complexity of emerging Automotive E/E Architectures"; Baden-Baden, September 25th, 2003

[39] AUTOSAR Development Partnership Information Pack – January 21st, 2003

[40] H. Heinecke, K.-P. Schnelle, H. Fennel, J. Bortolazzi, L. Lundh, J. Leflour, J.-L. Maté, K. Nishikawa, T. Scharnhorst "AUTomotive Open System ARchitecture – An Industry-Wide Initiative to

---

Manage the Complexity of Emerging Automotive E/E-Architectures" Convergence Transportation Electronics Association, 2004

[41] http://www.autosar.org

[42] E. Crawley; Systems Architecture class – Lecture 3; MIT SDM program; Fall 2004

[43] E. Fricke "Flexibility & Robustness – Major Platform Characteristics" Presentation at the Massachusetts Institute of Technology System Architecture class; December 5th, 2003

[44] F. Cela Diaz "An Integrative Framework for Architecting Supply Chains"; MIT SDM Thesis; October 2005

[45] http://www.bmwworld.com/technology/afs.htm

[46] C. Asam "Steer-by-Wire in cars"; Informationstechnisches Zentrum Technische Universität Clausthal

[47] http://dori2.technion.ac.il/opm/

[48] E. Crawley, O. de Weck, S. Eppinger, C. Magee, J. Moses, W. Seering, J. Schindall, D. Wallace, D. Whitney "The Influence of Architecture in Engineering Systems" MIT ESD, Engineering Systems Monograph; March 29th – 31st, 2004

[49] http://www.gm.com/company/gm_exp_live/events/paris_2002/gm/hywire/

[50] Discussion with William Taylor from Eaton; November 2005

[51] W. Taylor III "Supply-Chain Boundaries in Multi-Firm Product Design: A Multi-Industry Study" MIT SDM Thesis, January 2002

[52] C. Baldwin, K. Clark "Where Do Transactions Come From? A Perspective from Engineering Design" Saint Gobain Centre for Economic Research 5th conference; November 2002

[53] http://en.wikipedia.org/wiki/Suspension_(vehicle)

[54] http://inventors.about.com/od/astartinventions/a/air_bags.htm

[55] http://www.inc.com/magazine/20021001/24702.html

[56] http://www.whnet.com/4x4/abs.html

[57] http://www.engineerguy.com/comm/3812.htm

[58] http://en.wikipedia.org/wiki/Automatic_transmission#History_and_improvements

[59] http://en.wikipedia.org/wiki/Catalytic_converter

[60] http://www.thetech.org/nmot/detail.cfm?id=115&st=awardDate&qt=2002&kiosk=Off

---

[61] http://en.wikipedia.org/wiki/Robert_Stempel

[62] http://www.answers.com/topic/cruise-control

[63] http://powerelectronics.com/mag/power_top_power_milestones/

[64] http://en.wikipedia.org/wiki/Engine_Control_Unit

[65] http://en.wikipedia.org/wiki/Fuel_injection#Evolution

[66] http://en.wikipedia.org/wiki/Head-Up_Display

[67] http://en.wikipedia.org/wiki/OnStar

[68] Discussion with Jelani Ellington, ONSTAR; December 2005

[69] http://en.wikipedia.org/wiki/Power_steering

[70] http://www.rand.org/pubs/monograph_reports/MR614/MR614.appb.pdf

[71] http://en.wikipedia.org/wiki/Automobile_self_starter

[72] http://en.wikipedia.org/wiki/Continuously_variable_transmission

[73] C. Christensen "The HBR List, Breakthrough Ideas for 2004" Harvard Business Review

[74] R. Henderson, K. Clark, "Architectural innovation: the reconfiguration of existing product technologies and the failure of established firms - Technology, Organizations, and Innovation" Administrative Science Quarterly; March 1990

[75] E. Crawley; Systems Architecture class – Lecture 2; MIT SDM program; IAP 2004

[76] E. Crawley; Systems Architecture class – Lecture 1; MIT SDM program; Fall 2004

[77] Discussion with Timothy Thomas, Ford Motor Company; December 2005

[78] Discussion with Fernando Cela Diaz, fellow SDM; December 2005

[79] Discussion with Professor Thomas Keim, MIT; October 2005

[80] C. Baldwin, K. Clark, "Designs and Design Architecture: The Missing Link between "Knowledge" and the "Economy"" Harvard Business School; January 2005

[81] C. Christensen "When New Technologies Cause Firms to Fail" Presentation to the "Disruptive Technology" class, MIT SDM; March 2005

[82] Hart Research Associates "American's Views of Emerging Automotive Technologies"; June 2005

---

[83] D. Yoffie, M. Kwak “Judo Strategy – Turning Your Competitors’ Strength to Your Advantage” Harvard Business School Press; 2001