INTERNATIONAL DESIGN CONFERENCE - DESIGN 2006
Dubrovnik - Croatia, May 15 - 18, 2006.

DESIGN 2006

THE DESIGN OF SCIENCE BASED PRODUCTS: AN INTERPRETATION AND MODELLING WITH C-K THEORY

A. Hatchuel, P. Le Masson and B. Weil

Keywords: design theory, design space, science-based products, systematic design

1. Science-based products: mixing Research and Design

1.1 Science Based Products: an actionable definition

We define science-based products (SBP's) by two related design issues well recognized at the launching of the project: i) the product concept still requires *functional definition ;* ii) the product development requires *a program of scientific research* in order to explore the laws and properties of the *main phenomena* associated with the product. This definition implies the following distinctions:

- *An SBP is different from applying existing research results.* Applied research is usually considered as the application of existing scientific results coming from previous research to the design of some well identified functions.
- *An SBP is different from a basic science program.* Basic science program has usually no clear functional goal. SBPs clearly aim at new product, functional goals exist albeit only partially and in a broad form.

Examples of SBP's fitting with this definition : i) From industrial history : The design of the diesel engine [Bryant 1976] involved the exploration of new thermodynamics phenomena and the identification of innovative functions ; ii) To day, the design of new drugs in the pharmaceutical industry actually involves scientific research in chemistry, biology, biocomputer science, etc., with also wide-ranging debates around the functions (the disease to be cured may change with the scientific work) ; iii) The case of Cochlear implant development, as described by Van de Ven [Van de Ven, Polley, Garud, & Venkataraman 1999] , involved research in electronics, acoustics, physics, speech processing, etc. and the functional space was partly unknown (e.g. deep or partial deafness, more or less adaptable, more or less invasive).

The notion of SBP, as we define it, does not exist so clearly in design literature. Classical engineering design assumes that functions and phenomena are well-known and detailed requirements exist [Cross 2000; Jones 1992; Pugh 1991; Ullman 1997]. Functional definition is seen as important [Evbwuoman, Sivaloganathan, & Jebb 1996] yet it is not linked to the evolution of *research or explorations* made during in the project [Horton & Radcliffe 1995; Ottoson 2002; Tovey 2002].

1.2 SBPs and classic design theories: contrasting hypothesis

Having defined SBP's, our research issue can be clearly formulated: *how can we model a design process where functions and phenomena are both initially unknown?* Looking at how this issue is addressed in standard theories, we will show that classical design theories rely upon a knowledge base considered as given which contains functional, phenomenological and evaluation knowledge. Few design theories address the issue of *designing* the knowledge base itself as part of the design. Therefore, the design of SBPs will often be interpreted as a trial and error endeavour, an iterative, circular or random exploration process, which will be viewed as highly uncertain, risky and costly.

SBPs and systematic design : Classical systematic theory [Pahl & Beitz 1977] describes four main steps: clarification of tasks, functional design, conceptual design, embodiment design and detailed design. In this model, the SBP issue should be addressed in the second and third phase. When it comes to the third stage – where phenomenological language appears, several techniques are discussed to find principles solutions : selection of existing solutions (from literature or catalogues of solutions [Koller 1998],...) and generation of new solutions based on creativity, combination of pieces of solutions or model tests[Grabowski, Lossack, & El-Mejbri 1999; King & Sivaloganathan 1999; Rodenacker 1970]. Yet, since Rodenacker's addresses the *situation where new phenomena emerge during and thanks to the design process is not examined.* Systematic Design do not consider the integration of scientific research and misses an SBP feature: the impact of new phenomena on the generation of new functions.

Moreover Systematic design considers that an overall function can be divided into identifiable subfunctions corresponding to sub-tasks (Pahl & Beitz, First edition, chapter 2). This is not an acceptable hypothesis for SBPs:

- *In SBPs, the initial function is often very broad and will not necessarily be the main resulting function*. When designing an innovative UAV, the main function is not necessarily to fly without a pilot: an innovative UAV might be a UAV with a pilot in the loop (See [Sivaloganathan, Andrews, & Shahin 2001]).
- *Whenever the initial or main function is given, the decomposition in sub-functions is not pure functional reasoning*. It involves also knowledge on the underlying phenomena. Usually, functions should be described in a solution neutral form. Yet, in the case of SBPs, the functional decomposition interacts with the evolution of phenomenological knowledge. Hence, it is no more clear to define a „solution neutral" functional language as f*unctions may be a consequence of new physical or embodiment knowledge!.*

Other theories partially address the revision of the designer's knowledge base. Recent work on platform design introduces the idea of "a product functional model" [Du, Jiao, & Tseng 2003] where a modular architecture could be adapted to new information; similarly design templates" [Chandrashekaran, Stone, & McAdams 2004] or "platform based product architecture" [Du et al. 2003] are suggested. These models describe *families of embodiment strategies* which still need stabilized and frozen functional and conceptual design.

SBP and mapping-type theories : Suh axiomatic or Yoshikawa GDT both focus on well known mappings[Suh 1990; Yoshikawa 1981]. In Suh's axiomatic [Suh 1990] phenomena is captured by the equations linking the functional and the physical space. In SBPs these equations are partly unknown. Such case is viewed by Suh as a matter of creativity and designer expertise: it is not addressed by the theory itself**.** Yet, Suh is perfectly aware of SBPs situations and he already underlined the interaction between the functional space and the physical space : „*we must switch to the physical domain from he functional domain an vice-versa, in order to be able to proceed with the design process" (Suh, 1990, p. 36).* However he doesn't introduce it as a key element of the theory. he follows the standard view that functional diagrams need a domain specific knowledge [Kawakami, Katai, Sawaragi, Konishi, & Iwai 1996]. A similar pattern appears in GDT, one is supposed to use a „set of physical law concepts" that actually help to evaluate the feasibility of any entity concept. This means that this physical law concept set is considered as given in the process [Takeda, Veerkamp, Tomiyama, & Yoshikawa 1990]. However, one early author [Alexander 1964] foresaw this issue and criticized the functional, conceptual or physical stages suggesting that no predefined decomposition fit with the design process, which he viewed as a propagation of constraints : „*Each concept, at the time of its invention no more than a concise way of grasping many issues, quickly becomes a precept. We take the step from description to criterion too easily so that what is at first a useful tool becomes a bigoted preoccupation*" (p. 68). In other words, the language of exploration is neither pure functional, nor pure physical language and even not a simple combination of both (note that it doesn't mean that the *result* of the exploration can not be expressed in these languages; we will discuss this point later).

Now what about the design of SBPs? How can we model this process? Can we describe phases ? First, one can remark that the design of an SBP will *converge* towards a systematic design situation. If the design of SBPs cannot assume a stable functional and phenomenological space, it precisely regenerate such design languages. *But how to begin the design process without relying on these classical languages?*

1.3 The design of Science Based Products : Three central hypotheses

Based on this literature review, we can now formulate the three hypotheses of our research program:
P1: The design of SBPs requires to *reject the principle of predefined functions and phenomena (or physical solutions)*. Therefore, more powerful design theories are needed and we formulate the hypothesis that C-K design theory (Hatchuel and Weil 2003) *can account with rigor for the processes and reasoning that occur in the design of SBPs*.
P2: The design of SBPs requires several and interrelated learnings, but how does these learnings are organized and how are they related to design choices ? We define as a "design space", *a subset of the initial project that allows collective work where knowledge can be actually produced*. Thus, a design strategy for an SBP can be modelled as the selection of *a sequence of design spaces.* This sequence aims to (re)generate the languages of a systematic design process namely: a final functional space (F), a final phenomenological space (P), and a final validation space *(*L). Thus, from one design space to another *acquired knowledge can potentially inform these languages*: we name dF the variation of knowledge about F and similarly dP or dL. Hence, for each Design space i, there is a contribution (dFi, dPi, dLi) to the overall design. And the latter can be modelled as the sequence of these contributions.
P3: C-K theory gives a straightforward modelling and interpretation of the formation of design spaces and of their sequence. Thus, it is possible to understand how SBPs can be designed, including several learnings, suprises and redefinitions of both F, P, and L without generating a chaotic or completely uncontrolled process.

2. C-K theory and the design of SBPs : designing by design spaces

2.1 C-K theory of design : a brief overview

The C-K theory has been presented by Hatchuel and Weil [Hatchuel, Le Masson, & Weil 2004; Hatchuel & Weil 2003] and has been developed in several design contexts. It aims at a rigorous definition of Design reasoning (at the same level of modern set theory) and at a better understanding of Design organization and management in innovative projects. According to C-K theory, design needs two spaces and four operators:

- *The space of knowledge K*: a space of propositions that have a logical status, ie an attribute that defines the degree of confidence assigned to the proposition (in standard logic, propositions are true or false).
- *The space of concepts*, C, where a propositions ie. concepts have no logical status but are still interpretable in K (ie. all elements building the propositions in C come from K but do not belong to K).
- Design is the process by which a concept generates other concepts or is transformed into knowledge, ie propositions in K. Design seeks to expand concepts (δC) with existing knowledge (K) and to expand knowledge (δK) with existing concepts (C) through *disjunctive* (K→C) operators and/or *conjunctive* operators (C→K).
- The theory distinguished between two types of partitions. If the property added to a concept is already known in K as a property of one of the entities concerned, it is a restricting partition. If the property added is not known in K as a property of one of the entities involved in the concept definition, we have an expanding partition.

Fundamental results of advanced set theory warrant the consistency of C-K reasoning. (see Hatchuel and Weil 2003)

2.2 Proposition 1: C-K theory is adapted to SBP design situations

C-K theory meets all requirements coming from the definitions of SBPs: (i) C-K implies a learning process (□K, stimulated by C); (ii) contrasting with SD or axiomatic design, C-K theory doesn't rely on predefined languages: one can enter a C-K process without a functional definition or a stable phenomenological knowledge; (iii) C-K is a process theory.

2.3 Proposition 2: modelling a design space in C-K theory

- **Definition**: intuitively, a *design space* is a limited working context that allows learning within a design process. Formally, it is *a subset* of the design process, but *not all subsets are well formed design spaces*. Design spaces are shaped so that learning is possible including learning on *what has to be learned without taking into account all the attributes and knowledge available* in the design process.
- **Modelling design spaces** : This definition will be made clear with the C-K formalism as a design space can be defined as some C_0*-K_0* configuration with *a clear link* to an initial C_0-K_0 setting:
  - *C_0* is linked to C_0 by changes of attributes of the same entity*: C_0 being a generic proposition "entity x with properties $P_1 \ldots P_n(x)$", C_0* could be "entity x with properties $P_i \ldots P_j . P_1^* \ldots P^*_m(x)$" where Pi…Pj are properties chosen among $P_1 \ldots P_n$ and $P^*_1 \ldots P^*_m$ are new attributes, chosen to support the learning process.
  - *K_0* is a part of knowledge that can be specifically activated within a design space* (and is expected to be expanded). Thus K_0 - ($K_0 \cap K_0^*$) is the knowledge base that cannot be used by the designers inside the present design space. This might sound quite strange insofar as this means that a Design Space *restricts* the K-space to be explored.
  - The design process in C_0*-K_0* consists in a dual expansion ΔC_0^* (new attributes added to C_0*) and ΔK_0^* (new propositions added to K_0*).
  - The link between the overall C_0-K_0 setting and the design space is modelled by two types of *transition* operators. The first one are the operators going from C_0-K_0 to C_0*-K_0*, called the *design space building* operators (they have been described here above); the others are the *design space outputs extraction* operators. The latter consist in injecting parts of the ΔC_0^* and ΔK_0^* back into the main C_0-K_0 setting. They might consist in adding some new attributes to C_0 or in adding new pieces of knowledge in K_0.

2.3.1 An example of Design Space: designing a new UAV without studying airworthiness

UAVs (Unmanned aerial vehicles) are usually SBPs. In one project, C_0 was "an innovative and airworthy UAV". Yet, the first design space was defined as „an autonomous helicopter for traffic surveillance" with a research focus on artificial intelligence and computer vision. Let's model the formation of this design space:

- C_0: "x=a flying vehicule", P_1 = "airworthy", P_2 = "unmanned", P_3 = "innovative".
- K_0: all knowledge available and producible.
- C_0*: withdraw P_1 and adds P_4 ="being a helicopter" and P_5 = "the mission being for traffic surveillance".
- K_0*: all knowledge on aircraft, military missions, and control is not considered intentionnally! Why ? *Traditional UAVs designs are based on control as a main discipline (control over long time flight, control ultra-light objects,…) that determines both functional and phenomena language*. The design space explicitly excluded the control knowledge to explicitly organize the learning process on two underestimated disciplines in the UAV business: artificial intelligence (how can an object "decide" or deliberate" in front of an original situation) and computer vision (what are the tools that can scan and analyse an environment).
- Validation in C_0*-K_0*: in this design space validation was linked to these disciplines and airworthiness was not yet considered!

2.3.2 The properties of a design space: restricting design to a designable and informative concept

Defining Design spaces is necessary to SBPs as the properties of a design space actually allow to avoid the obstacles that are constitutive of SBPs:

- *When a well-formulated concept is missing*, it is possible to begin the exploration on a derived concept.
- *When knowledge is missing*, it is possible to add some knowledge (by research); when there are too many learning opportunities, one can focus learning on specific areas.
- *When validation operators are unavailable*, it helps to build a design space where some validation becomes possible.

Hence in the case of SPBs, building design spaces is the unique *strategy*. Functional and phenomenological spaces are partly unknown in SBP's *but one cannot learn about these spaces without entering the design process.* Conversely, Design validation is impossible without functional and phenomenological validation. Building a Design space allows to avoid this circular trap. It allows simultaneously to design and to learn by *a restriction that will allow for new learnings* (on F, P, and L). This allows a completely different view of the design process where each design space is *F-P-L complete,* and depends on the results of the previous one.

2.4 Proposition 3: modelling a sequence of (F, P, L) learnings with C-K theory.

Now we can model an SBP sequence of design spaces in the following way:

- *At the upper level we model the evolution of the overall C_0-K_0 expansions* and the formation by restriction on C and K of Design spaces and their (dF, dP, dL) contributions.
- *At the lower level, we represent each design space i* as a specific C_i*-K_i* setting, each representation also describing C and K expansions.
- The design space building operators control the top-down transitions; the output extraction operators control the bottom-up transitions.
- Theoretically, the *whole process remains a C-K expansion* but isolating the design spaces actually explains how SBPs design is made *actionable* while the whole process keeps its consistency.

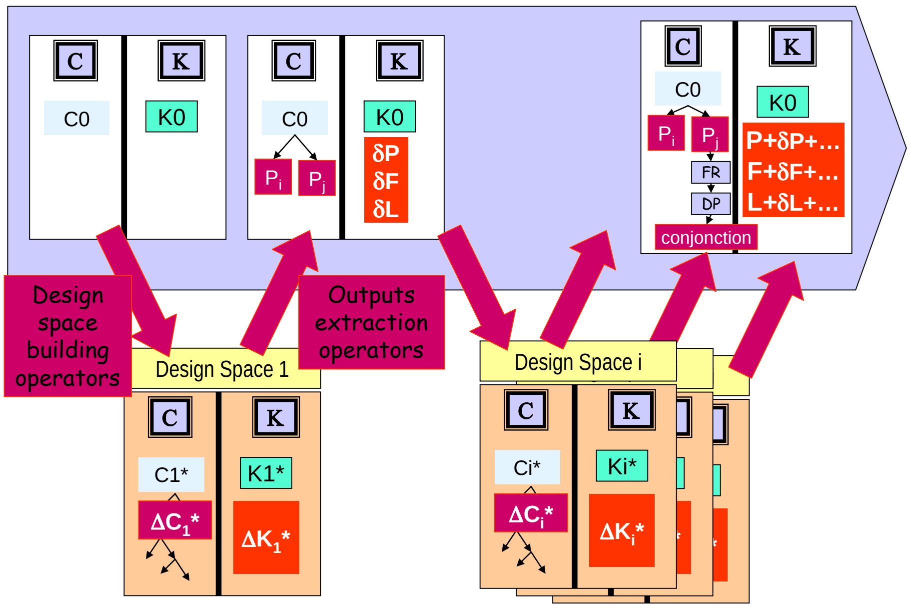

Figure 1. The SBP design process

2.5 A new model for organizing the SBP design process

What does prove that the sequence of design spaces is convergent? Nothing can prove it in absolute terms. But within C-K theory, it is easy to prove, that the existence of a conjunction from C→ K, means that there exists at least one (Fi, Pi, Li) that can be interpreted as one SBP product. The existence of a conjunction means that it exists a sequence of attributes P1,P2,…Pn so that the proposition C: "”It exists x, P1(x), P2(x), Pn(x)” is true in K” is true. Now some of these Pi can be interpreted as Fi, others as Pi ; and the proposition Cf is true is possible only if some Li allows to validate it in K[1].

Let’s also mention the organizational consequences of this framework. We have underlined that a systematic design process, based on stabilized languages, can be sequential, convergent and exploratory in limited areas. By contrast, an SBP might appear as iterative, risky, and random. However, within the C-K framework, design spaces, transitions, and learning processes makes perfectly sense and, by the way, the Design becomes “rational”: meaning here that each exploration is clearly situated and controlled within the corresponding C-K expansions. Therefore, the design of SBP’s can managed by monitoring the formation of design spaces and the transitions from one to the other (Hatchuel et al. 2005)

3. Modelling the design spaces: a case study of designing a new “bio-climate” in cars

To illustrate these theoretical results on an industrial case, we show : (1) how C-K modelling accounts for the explorations in this industrial situation, (2) how it helps to understand the main design spaces of the process, (3) and how it enables to monitor the exploration progress. Finally, C-K theory reveals the rationale and sequentially of the design process of an SBP, that was apparently chaotic and iterative.

3.1 Designing the smells of a car: a seemingly chaotic pattern

In the late 90s, a European car manufacturer was investigating a broad innovation field: car olfactory signature. This is an SBP: what are the main phenomena behind smells and olfaction, particularly in car? What are the potential functions?

In the classical languages of engineering, the design process followed by this project appears as chaotic, costly and risky. Instead of a fluid transition between functional, conceptual and embodiment design, we find a first trial around one small function (add a smell in a car), then the formulation of a broad innovative concept (bioclimate), then one test on a very focused property (air tightness) and in parallel the works and studies on an exploratory prototype to formulate precise functions, as if functions would come in the end of the process! In this context, let’s illustrate our framework.

3.2 Design space #1: “add a smell” and its learnings

Transition from C_0-K_0 to C_1*-K_1*. The initial C-K configuration is characterized as follows:

- K_0 = knowledge on car design; some knowledge on smells and recent crisis on smells in competitors’ car models.
- C_0 = One concept is formulated: a “car olfactory signature” (this is a concept: no logical status but interpretable proposition in K).

Designing at the upper level begins: why not add smells in a car, to get a “brand olfactory signature” or to play a “fragrance organ”? But it rapidly stops: how to work further on such an issue?

The first design space emerges: It is suggested that one designers team investigate a restriction of the initial concept (add smells in car), not to get a solution but to find the main phenomena and functions around smells in car. This opens the first design space, defined by the following transition operators:

- From C_0 to C_1*: renounce on a lot of properties on “car” in the industrial sense; focus the exploration on smells in car and not at the level of “olfactory signature”.

[1] The rigorous proof of this result need a complete presentation of the mathematical foundations of C-K theory. This will be given in forthcoming papers.

- From K_0 to K_1* = don't consider all knowledge on car manufacturing or car engineering constraints.

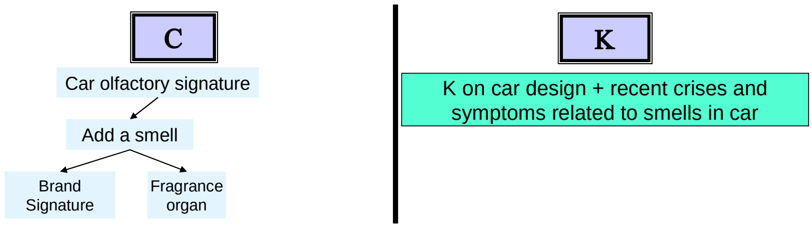

Figure 2. Initial configuration – launching design space #1 "add a smell in car"

Designing into C_1*-K_1*: In this design space #1, the analyse of the existing systems of perfume diffusion revealed one big issue in olfaction: perception. Perception is different if the car interior is "free of smells" or is a complex smell ground. This led to learn on car smell sources: plastic material description, air-conditioning and exterior air, passengers and luggage,… and the way to control them to get a "neutral" smell ground (filters, new materials…). This finally led to develop a *full model of smells* in car, identifying smells sources, smells flows, smells media and smells control variables like temperature, volatile compounds, air speed and hygrometry. It also showed that perfume diffusion wasn't perceptible in case of a complex smell ground.

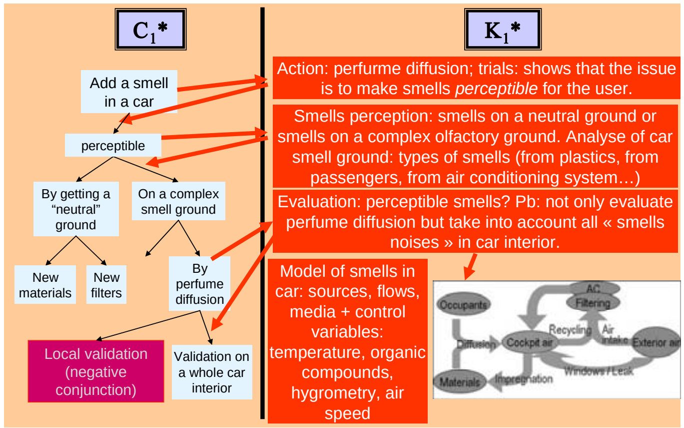

Figure 3. Design Space 1: designing the functional, phenomenological an validation languages

Transition from C_1*-K_1* to C_0-K_0: learning on F, P, L and C_0-K_0 expansion. After five months of enquiry, the designers went back to the overall SBP design process. Are they ready for a systematic design process? The overall knowledge base is enriched:

ΔP	olfactory perception, models for smells in car interior with identified action means, links with volatile compounds, temperature, aird speed and hygrometry, flow model…)
ΔF	olfactory neutrality, fight against bad smells, links wit other functions like thermal comfort, depolution,…
ΔL	it is known that no validation can be done locally but should include a whole car interior.

Hence the validation knowledge is still insufficient to organise a systematic design process: From this knowledge it is possible to derive several alternatives from the concept "smells under control n a car", each alternative being structured as a partial systematic design: *function first and design parameter afterwards*; but this design is still *partially* systematic, insofar as it is impossible to validate the design parameter efficiency (see "smells under control" on figure 4).

Consequently the overall C-K configuration is still an SBP. *Knowledge acquired in the design space is used to expand C_0-K_0.* The knowledge on the car smells phenomenology suggests that "smells in car" is just a partition of a broader concept around the "car climate" (this is how a restriction on a design space can help to formulate broader and more original concepts). The bioclimate concept can be treated by keeping the classic airflow architecture (see "control smells", studied in the first design space; see also alternatives on car interior pollution); but the concept can also be treated by changing the architecture. The airflow model then immediately suggests several alternatives: explore new airspeed regimes (ventilation and air-conditioning today require high speed, which implies turbulent regimes in car interior), identify, separate and differentiate several airflows (per passenger for instance); complete the airflow with added controlled gaz sources (see figure 4).

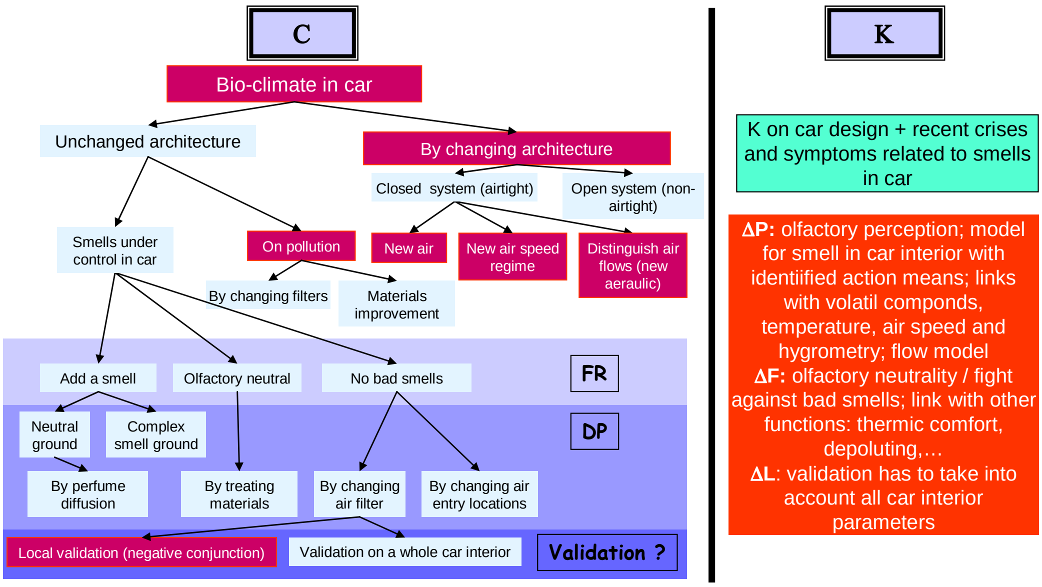

Figure 4. Back to the main C-K: i) declining functions and design parameters to partition "smells under control". Ii) formulating a broader concept and its first partitions

Comments on this first design space exploration:

- In the design space, the exploration doesn't follow a systematic design process.
- The design space rely on a strong restriction: it excludes large knowledge areas (for instance: costs, feasibility, olfactory brand design)
- The design space is different from a basic research program. The latter would have only addressed smells phenomenology (not even smells in car), whereas the design space explored simultaneously the phenomena on smells in car, the function (a perceptible smell) and the validation techniques.

3.3 Design spaces #2 and #3: testing air-tightness and prototyping mild air conditioning

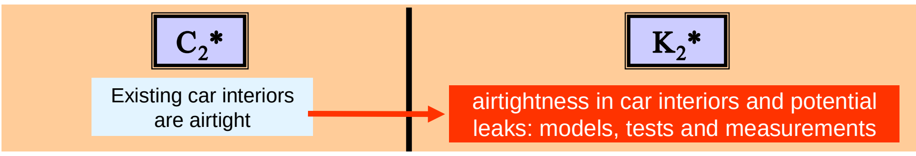

Figure 5. Design Space 2: testing airtightness

Transition from the overall C-K to C_2*-K_2* and C_3*-K_3*. The main C-K tree suggests at least two ways: first the flow models and the alternatives are strongly different if the car interior is airtight or not. As a consequence it appears necessary to have a quick and thorough analysis on car interior airtightness:

- C_2* = "existing car interiors are airtight";
- K_2* = knowledge to test car interior airtightness (bench test).

Design space #2 is just a test of the proposition C_2*; this test provokes a knowledge expansion on tests for interior air-tightness.

The "bioclimate" opens several architectural alternatives. How to explore them quickly? It is proposed to have a *prototype combining two solutions*: a new air speed regime with separate airflows (design space 3).

- C_3* = "vehicle with vertical airflows, for thermal comfort, through mild air conditioning";
- K_3* = previous knowledge on car engineering (and smells!) largely frozen. Add knowledge on modelling thermal phenomena and a device to produce knowledge, a focus group.

The design process will be limited to a couple of months and involved a focus group (as a K→K operator). The design space helped to identify two types of thermal comfort: mild air conditioning and an hybrid air-conditioning with two air speed regimes, the first one for fast temperature transitions and the second one for permanent regimes. It also enabled to model cooling capacity with vertical airflows.

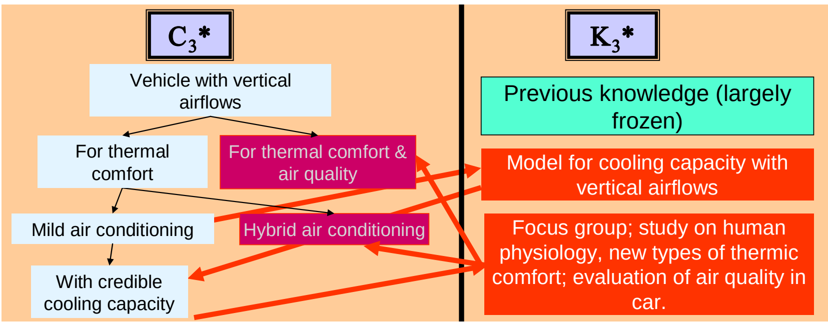

Figure 6. Design Space 3: functional explorations from a "phenomenological" prototype

Transition from C_2*-K_2* and C_3*K_3* to the main C-K: learning on F, P, L. Coming back to the main C-K after these two design spaces, we see that these explorations led to the enrichment of the functional, phenomenological and validation bases.

ΔP	flow modeling; credible cooling capacity; vertical airflow; human physiology; air pollution
ΔF	new types of thermal comfort (transitional / permanent) ; air quality
ΔL	validate through cooling capacity models; focus group techniques; applying standard development criteria to the prototype leads to negative conjuntion

We find a *new systematic design structure* under one alternative of the bio-climate ("airtight architecture with vertical airflows"). This is a major result: we now dispose of a systematic design basis to cope (partially) with concept that was at the beginning an SBP!

Comments on the whole process:

- The C-K formalism helps to give a much clearer picture of the steps, the alternatives, the progress, the status of each focused exploration.
- The exploration was driven by three main design spaces. Each design space was a *wellspring for learning, based on strong restriction of the main C-K.*
- The variety of design spaces (a conceptual exploration (add a smell), a test and a prototype) underlines that the design process doesn't follow a "specification process" where spec precision is enhanced at each step, detailed design and tests being the last steps. For example

design space #2 (testing car airtightness) shows that SBP might require in early phases thorough analysis on design details.

- Each design space contributed to generate the languages P, F and L, building a strong base for future business development around bio-climate in car in systematic design.
- Finally the C-K language contributed to make visible the underlying organization. Far from a trial and error process, we see an organized cumulative exploration process with two types of activities, learning phases in design spaces and value management and capitalisation at the main C-K level. This process is far from the systematic design linear process (based on stabilized languages) or from a two-step model ("research then systematic design").

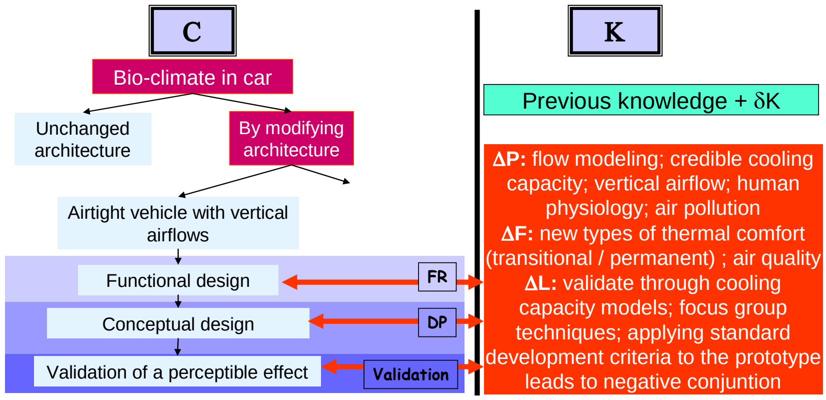

Figure 7. Back to the main C-K: embryon of a systematic design for a new business

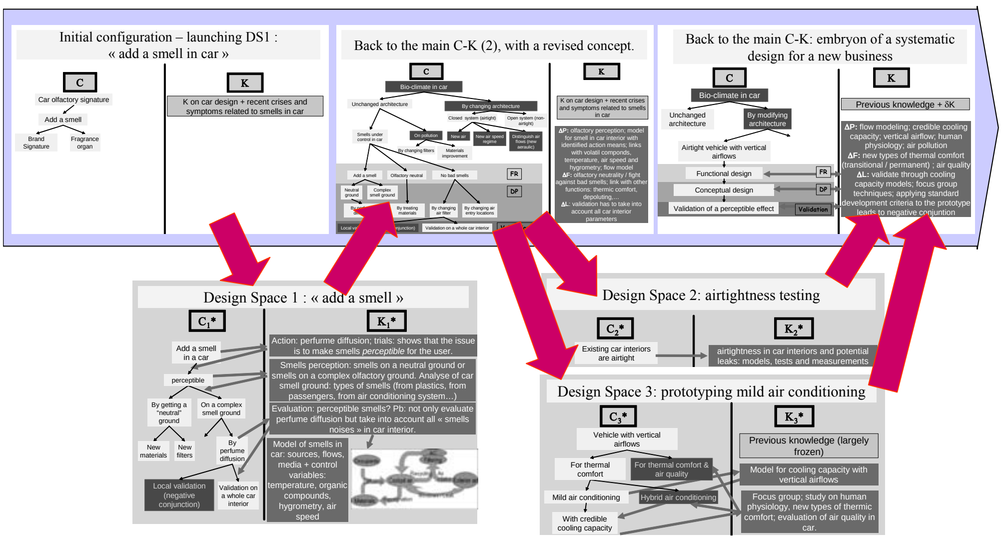

Figure 8. The overall SBP design process in the design space language

4. Conclusion and further research

In this paper we studied a specific class of design problems: the design of science based products. We have shown that this class of problem can't be addressed with classical design theories, which are actually relying on a given base of functional, phenomenological and evaluation languages, ie the *languages that have to be designed in an SBP situation*. We have shown three main results: the C-K theory of design meets the requirements for addressing SBP design situations; a design strategy for an SBP can be modelled as the selection of *a sequence of design spaces* aiming at (re)generating the languages of a systematic design process namely; C-K theory gives a straightforward modelling and interpretation of the formation of design spaces and of their sequence. Organizationally this means that the C-K formalism provides a new language to organize linear design processes in SBP instead of the seemingly unavoidably chaotic processes. The car bio-climate case illustrated how C-K helped to interpret an SBP design process.

Case and theory paved the way to further research:

- C-K was used to interpret the actors reasoning. It could also help to identify the design spaces: is it possible to design a design space? To compare between different potential design spaces?
- It would be interesting to try to identify different types of design spaces and different types of transition operators. How kind of typology could be built?
- We have given one example of a global SBP design process. What are the main features of such a process? What is the economics of learning in such a process?

References

Alexander, C. "Notes on the Synthesis of Form" Harvard University Press, Cambridge, MA, 1964.

Bryant, L., "The Development of the Diesel Engine", Technology and Culture, 17 (3), 1976 pp 432-446.

Chandrashekaran, B., Stone, R. B., & McAdams, D. A., "Developing design templates for product platform focused design", Journal of Engineering Design, 15 (3), 2004 pp 209-228.

Cross, N. "Engineering Design Methods. Strategies for Product Design." John Wiley & Sons Litd, Chichester, England, 2000.

Du, X., Jiao, J., & Tseng, M. M., "Modelling platform-based product configuration using programmed attributed graph grammars", Journal of Engineering Design, 14 (2), 2003 pp 145-167.

Evbwuoman, N. F., Sivaloganathan, S., & Jebb, A., "A Survey of Design Philosophy, models, methods and systems", Journal of Engineering Manufacture, 210, 1996 pp 301-320.

Grabowski, H., Lossack, R.-S., & El-Mejbri, E.-F."Towards a Universal Design Theory". Proceedings of Integration of Process Knowledge into Design Support Systems, Kals, H., & Houten, F. v., Kluwer Academic Publishers, University of Twente, Enschede, The Netherlands, 1999, pp 47-56.

Hatchuel, A., Le Masson, P., & Weil, B."C-K Theory in Practice: Lessons from Industrial Applications". Proceedings of 8th International Design Conference, Marjanovic, D., Dubrovnik, 18th-21st May 2004, 2004, pp 245-257.

Hatchuel, A., Le Masson, P., & Weil, B., "The Development of Science-Based Products: Managing by Design Spaces", Creativity and Innovation Management, 14 (4), 2005 pp 345-354.

Hatchuel, A., & Weil, B."A new approach of innovative design: an introduction to C-K theory". Proceedings of ICED'03, august 2003, Stockholm, Sweden, 2003, pp 14.

Horton, G. I., & Radcliffe, D. F., "Nature of rapid proof-of-concept prototyping", Journal of Engineering Design, 6 (1), 1995 pp 3-16.

Jones, J. C. "Design Methods" John Wiley & Sons, Inc., London, 1992.

Kawakami, H., Katai, O., Sawaragi, T., Konishi, T., & Iwai, S., "Knowledge acquisition method for conceptual design based on value engineering and axiomatic design theory", Artificial Intelligence in Engineering, 1, 1996 pp 187-202.

King, A. M., & Sivaloganathan, S., "Development of a Methodology for a Concept Selection in Flexible Design Strategies", Journal of Engineering Design, 10 (4), 1999 pp 329-349.

Koller, R. "Konstruktionslehre für den Maschinenbau. Grundlagen zur Neu- und Weiterentwicklung technischer Produkte mit Beispielen." Springer, Berlin, 1998.

Ottoson, S., "Virtual Reality in the Product Development Process", Journal of Engineering Design, 13 (2), 2002 pp 159-172.

Pahl, G., & Beitz, W. "Konstruktionslehre (english title: engineering design)" Springer Verlag, édition anglaise: The Design Council, Heidelberg, version anglaise: London, 1977.

Pugh, S. "Total Desgn. Integrated Methods fo Successful Product Engineering" Prentice Hall, Pearson Education., Harlow, England, 1991.

Rodenacker, W. G. "Methodisches Konstruieren" Springer Verlag, Berlin, 1970.

Sivaloganathan, S., Andrews, P. T., & Shahin, T. M. M., "Design Function Deployment: a Tutorial Introduction", Journal of Engineering Design, 12 (1), 2001 pp 59-74.

Suh, N. P. "Principles of Design" Oxford University Press, 1990.
Takeda, H., Veerkamp, P., Tomiyama, T., & Yoshikawa, H., "Modeling Design Processes", AI Magazine, Winter 1990, 1990 pp 37-48.
Tovey, M., "Concept design CAD for the automotive industry", Journal of Engineering Design, 13 (1), 2002 pp 5-18.
Ullman, D. G. "The Mechanical Design Process" Mc Graw Hill, Boston, MA, 1997.
Van de Ven, A., Polley, D. E., Garud, R., & Venkataraman, S. "The Innovation Journey" Oxford University Press, New-York, Oxford, 1999.
Yoshikawa, H. "General Design Theory and a CAD System". In Sata, & Warman (Eds.), "Man-Machine Communication in CAD/CAM, proceedings of the IFIP WG5.2-5.3 Working Conference 1980 (Tokyo)": 35-57. Amsterdam, North-Holland, 1981.

Armand Hatchuel
CGS & the Fenix program at Chalmers University
Ecole des Mines de Paris
60 boulevard Saint Michel
F-75 272 Paris Cedex 06, France
Tel.: +33 1 40 51 92 21
Fax.: +33 1 40 46 03 01
Email: hatchuel@ensmp.fr