



HAL
open science

Uncovering the specificities of CAD tools for industrial design with design theory – style models for generic singularity

Pierre-Antoine Arrighi, Pascal Le Masson, Benoit Weil, Akin Kazakçi

► To cite this version:

Pierre-Antoine Arrighi, Pascal Le Masson, Benoit Weil, Akin Kazakçi. Uncovering the specificities of CAD tools for industrial design with design theory – style models for generic singularity. *International Journal of Design Creativity and Innovation*, 2016, 10.1080/21650349.2016.1190301 . hal-01389651

HAL Id: hal-01389651

<https://hal.science/hal-01389651v1>

Submitted on 28 Oct 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

"UNCOVERING THE SPECIFICITIES OF CAD TOOLS FOR INDUSTRIAL DESIGN WITH DESIGN THEORY - STYLE MODELS FOR MASS SINGULARITY."

To cite this article: **Arrighi, P.-A., Le Masson, P., Weil, B., and Kazakçi, A. (2016).**

“Uncovering the specificities of CAD tools for industrial design with design theory – style models for genericsingularity.” *International Journal of Design Creativity and Innovation*, pp. 1-19.

Authors

Dr. Pierre-Antoine Arrighi^{1, 2},
Prof. Pascal Le Masson¹,
Prof. Benoit Weil¹,
Dr. Akin Kazakçi¹

Affiliations:

¹ MINES ParisTech – PSL Research University
CGS-i3 UMR CNRS 9217
60 boulevard Saint Michel, 75006 Paris, France.

²DassaultSystèmes

Corresponding author:

pascal.le_masson@mines-paristech.fr

Tel: +33 1 40 51 92 21

Uncovering the specificities of CAD tools for industrial design with design theory – style models for generic singularity

Abstract

According to some casual observers, computer-aided design (CAD) tools are very similar. These tools are used to design new artifacts in a digital environment; hence, they share typical software components, such as a computing engine and human-machine interface. However, CAD software is dedicated to specific professionals—such as engineers, three-dimensional (3D) artists, and industrial designers (IDs)—who claim that, despite their apparent similarities, CAD tools are so different that they are not substitutable. Moreover, CAD tools do not fully meet the needs of IDs. This paper aims at better characterizing CAD tools by taking into account their underlying design logic, which involves relying on recent advances in design theory. We show that engineering CAD tools are actually *modeling* tools that design a *generic variety* of products; 3D artist CAD tools not only design but immediately *produce* single digital artefacts; and IDCAD tools are neither a mix nor an hybridization of engineering CAD and 3D artist CAD tools but have their own logic, namely to create new conceptual models for a large variety of products, that is, the creation of a unique original *style* that leads to a *generic singularity*. Such tools are useful for many creative designers beyond IDs.

Keywords: computer-aided design (CAD); industrial designers; engineers; 3D artists; conceptual models

1. Introduction

This paper is based on a dual motivation. First, there is an industrial motivation. It is now well known that design processes should integrate different design professionals, such as industrial designers, engineers, and graphic designers. It is also well known that this integration must be supported by contemporary computer-aided design (CAD) tools that are currently at the basis of almost every design process. However, at the present time, every profession has, to some extent, its own CAD tools, with only limited compatibility. The question is whether it is possible to integrate these tools. Initially, all these professional CAD tools appear very similar; in the literature on CAD, the focus is mostly oriented toward the technologies and algorithms for generating the artifact (Stergiopoulos et al. 2003; Dyn et al. 2009; Mandil et al. 2011; Bodein et al. 2012). Despite these apparent similarities, there are great discrepancies in professional CAD development. Engineering CAD and three-dimensional (3D) artist CAD are well established and improve continuously whereas, in a type of blurred middle ground, industrial designers (IDs) are at the center of a great number of proposals and innovative tools (e.g., Rhino3D and Alias), but there are still debates about the inadequacy that IDs encounter when they try to use CAD tools (Tovey 2002; Dorta et al. 2008). IDs tend to use tools that were not made for them and consequently struggle to maintain their creativity and be integrated into industrial processes. Hence, it is difficult to fully understand the specificities of ID CAD compared with engineering CAD and graphical CAD. Moreover, it is unclear how these different tools might be made compatible.

The second motivation for this paper relates to design science and design theory. Recent advances in design theory provide us with an ecology of theories that might be powerful enough to bridge the gap between different professional approaches to design (Le Masson et al. 2013). However, research on CAD tools very rarely builds an analytical framework on design theory. Many research works actually use more generic notions, such as “boundary objects” or “artifacts,” which are inherited from sociology or management, that tend to hide the specificities of design reasoning and barely account for the variety of CAD tools and the incompatibilities that were recorded in the use of these tools by professionals.

Given these two motivations, this paper aims to account for the design logic at the root of CAD tools used by engineers, 3D artists, and IDs by using design theory to model the reasoning steps of designers using these tools. More precisely, this paper intends to understand the specificities of ID CAD in comparison with engineering CAD and 3D artist CAD.

To summarize, this paper clarifies:

(1) **The differences in the missions of professional CAD tools:** engineering CAD supports the design of product models so that they are compatible with a “physical” environment and generic conditions; 3D artist CAD tools support the design of a single object to be included in a “virtual” world; and ID CAD tools support the design of a *singularity for genericity*—that is, a singular original style to be applied in generic conditions.

(2) **The design logic behind each professional CAD tool:** engineering CAD relies on modeling, that is, finding a parametric product *model* by combining existing design parameters (DPs) to address existing functional requirements (FRs). 3D artist CAD relies on the immediate design of the object, similar to sculpting or making art more generally. It directly creates the virtual object and is not a “representation” of something that would exist somewhere else. Instead, it is a “presentation”—the object is “made present” by the 3D artist using CAD. Finally, ID CAD aims at *designing styles models*, which is very specific and different from modeling (i.e., fitting a

general model to a particular environment) and art making (which does not necessarily require models).

We start by identifying the tendencies of the literature when analyzing such tools. We formulate a generic framework to model the reasoning of CAD tools based on design theories. We then use our framework to analyze the following:(1) how it accounts for the differences between contemporary professional CADs (synchronic approach);and (2) how it accounts for the long-term evolution of professional CAD software (diachronic approach).

2. Literature review

The literature clearly distinguishes between three groups of professionals whose CAD tools: (1) engineers who work in the manufacturing industry and use CAD tools to design industrially produced physical objects; (2) 3D artists who work in the entertainment industry and use CAD tools to design virtual environments and content for animation, movies, and video games; and (3) IDs who work in industry and concentrate on the user experience of manufactured products. For each of these groups, the literature in computer science, engineering design, and design creativity is distributed along the following two axes:(1) the modeling capacities of the geometric kernels, which define what type of objects can be created; and (2) the tools' interfaces, that is, man-machine interfaces (MMI).

2.1 CAD tools for engineers

The first identified category in CAD tool literature is engineering CAD tools, such as SOLIDWORKS, Pro/ENGINEER, AutoCAD, and NX. Over 20 years ago, a special issue of *Research in Engineering Design* (Rosen and Peters 1993) underlined the paradigm shift in CAD from drafting to product data modeling. It proposed hypotheses for the future of engineering CAD modeling based on the emergence and potential generalization of “features” in CAD (Rosen 1993). According to these hypotheses, a common set of subfeature element types, element–element relations types (spatial, logical, etc.), and design features associated to the subfeatures and relations can: enable performance, cost, and other lifecycle analyses; embody physical principles; and enable checks for consistency with design principles. For example, (Shah and Rogers 1993) developed concepts for product modeling with a uniform set of structures and relationships, and developed approaches to derive the assembly location from high-level specifications. It appeared that no single feature set could satisfy all engineers, so the “features” used in modern CAD are only a limited approximation of the CAD tools based on a feature-based representation as imagined by Rosen et al.¹ Retrospectively, it appears that this special issue was quite visionary in two aspects:

- (1) It insisted on the modeling aspect of CAD, where a CAD tool finds a model (for example, of an object, parts, components) that can be considered as “satisfying” the composition of “features” to meet requirements.
- (2) It identified a trend in CAD in that the design of new relations was not limited to geometric relations, as in parameterized drafting, but could be extended to relations between several types of entities and, in particular, functional entities.

This was confirmed by subsequent research: for a summary, see (Zeng and Horváth 2012). For instance, works on CAD modeling aim to integrate conceptual design (Brunetti and Golob 2000); extend parametric design to complex products, such as cars (Salehi and McMahon 2011); support constraint identification for more efficient optimization logic (Bettig and Shah 2001); enable modeling with heterogeneous solids (Liu et al. 2004a); work on a language of part modeling to enable higher compatibility (Xu et al. 2013); or create multi-level assembly models for CAD (Chen et al. 2012).

This modeling logic structures the computer science literature on engineering CAD, which is oriented toward improving the geometric model kernel and interfaces of the tools.

2.1.1 Geometric model kernel for engineering CAD tools

The geometric model kernel is the engine of the CAD tool. It determines what type of objects can be modeled and managed by the software during the design process. This category of the literature is globally evolving toward more functions and the ability to solve more problems of various natures, while being able to handle new classes of objects. It takes into account cognitive efforts of designers (Nguyen and Zeng 2014). As shown in (Liu et al. 2014), CAD systems aim to “recognize, interpret and process human emotions” to integrate the potential bias that can arise in the process of designing a concept that is “right first time.” Some scholars, such as (Mandil et al. 2011), modify the geometric kernels to describe new attributes, such as geometric dots and line segments, to depict early design-stage mechanisms. (Dyn et al. 2009) proposed replacing uniform parameter values by chordal and centripetal values to achieve nonlinear schemes that are invariant under solid body and isotropic scaling transformations. These papers have shown the progressive implementation of capacities for the modeler to handle more types of objects, such as functional specifications depicted geometrically. The functions and capacities of the software are also affected by the ability to manage more constraints, some of them of a different and totally new type, such as user preferences (Kelly et al. 2011). CAD can even be used to “automate the procedure of customization” while using standardized components (Kwok and Wang 2014). Because of increasing data and knowledge generated during the design and manufacturing processes, (Ouertani et al. 2011; Bodein et al. 2012) proposed a new framework for the exploitation and curation of knowledge. (Goel et al. 2012) summarized these trends by stating that future CAD should be “cognitive, collaborative, conceptual and creative,” where “creative design” is actually a capacity to access “systems functionally and at different levels of abstraction,” remaining in a “search” paradigm where the designer looks for solutions to a given problem (p. 897).

2.1.2 Interfaces for manipulating digital engineered content

Interfaces enable interaction with machines. The growing demand for experiencing products at the early stages of design has created a strong need for new interfaces with extended sensory-motor capacities. There is a progressive shift from old interfaces, such as the mouse and keyboard, toward immersive ways of interacting, such as touch, multi-touch, and virtual reality, to propose enriched interactions (Stergiopoulos et al. 2003; Deschênes et al. 2004). (Kang et al. 2013) focused on the step of transferring content to new portable devices, that is, tablets and smartphones with multi-touch. (Liu et al. 2004b) proposed a system for direct interaction and modification with a haptic device (see also Fuge et al. 2012).

To summarize, published papers on CAD tools for engineers seek to *increase the capacities of geometric modeling kernels to handle more functions, algorithms, and properties of the underlying design concept throughout the entire design process*, which includes the factory's design, manufacturing, and maintenance. The digital object tends to become the most perfect model of the future real product, up to the chosen tolerances. Furthermore, the literature aims to increase interactions between the user and model, for example, with immersive technologies.

2.2 CAD tools for 3D artists

The second identified category in the literature is artistic CAD design tools, such as 3ds Max, Maya, and ZBrush. 3D artists produce 3D models required for video games, animated films, and special effects in movies, known as computer-generated imagery (CGI). This family of products evolves within a galaxy of other services, such as animation, rigging, texture mapping, unfolding, and rendering. The literature again stresses two aspects: the geometric model kernels and the interfaces for manipulating them.

2.2.1 Geometric model kernel of artistic CAD tools

The geometric model kernel for artistic CAD tools is usually designed to manage the highest number of polygons. The more polygons, the greater the density of detail that can be achieved. In most cases, artists need tools that allow the direct manipulation of the rendering result. The most advanced technique in this domain is generically called voxel, and pixolin ZBrush terminology. New models have been introduced to address the specific needs of the entertainment industry, such as video gaming, which often encounters the issue of computing collisions inside real-time environments (Mishkinis et al. 2012). (Loop and Schaefer 2008; Loop et al. 2009) introduced a new method of approximating subdivision surfaces, which are the most used modeling technique of 3D artistic CAD tools, with hardware accelerated parametric patches, which improve the memory bandwidth requirements for patch control points. This allows an extension of the number of polygons displayed by the computer to add more detail to the model. (Kowalski et al. 1999) proposed building on the ability of artists and illustrators to evoke the complexity of fur or vegetation with relatively few strokes on the boundaries of objects. This reduces the total number of polygons while increasing the richness and density of detail. Walt Disney Studios recently published the process it used for its Oscar-winning animated short film *Paperman* in 2012. In their SIGGRAPH talk, Whited et al. (Whited et al. 2012) explained how a hand-drawn appearance can be achieved with computer-generated models to make final animated movies look similar to old animated movies that were drawn by hand.

2.2.2 Interfaces for manipulating artistic digital content

Software designers and researchers have proposed new types of interfaces specially designed for 3D artists. The most visible strategies are the imitations of the sketch and clay modeling designing steps. For example, as early as 1996, (Gross and Do 1996) proposed a system that could recognize and interpret drawings. (Igarashi et al. 2007; Nealen et al. 2007) worked on a project for creating 3D shapes from sketched two-dimensional (2D) input. This technique allows the user to add, remove, and deform control curves, which then generate free-form surfaces. The curves can have arbitrary topologies and do not need to be connected to one

another, which allows the system to manage both curve deformation and the subsequent surface optimization. Other researchers, such as (Keefe et al. 2001), have explained how new 3D immersive interfaces can be used to generate artistic 3D content. They proposed an artistic medium that uses a 3D analog of 2D brush strokes to create 3D works of art in a fully immersive environment. (Zhang et al. 2009) proposed a virtual clay modeling system by which users can directly manipulate the shape of a virtual object as they can do with a clay model.

In papers on CAD tools for 3D artists, researchers have sought methods that increase the capacities of geometric modeling kernels to optimize real-time and recalculated visualization with the best resolution possible. This focus is required because the design exists and evolves *inside a virtual environment* for which it must be optimized. Designers want to make the model plausible inside its environment. The profusion of details creates a backstory, past, and context. Conversely, the literature proposes interactions with rich interfaces that have a strong focus on sketch-based interfaces and clay modeling metaphors. They imitate the artist's studio and ancient techniques. Natural and direct interfaces are intended to help the designer control an appearance and a final rendering result so that the model can blend within the virtual environment.

This first review shows how similar issues—such as modeling kernels and interfaces—are addressed differently by the literature when considering engineering CAD or artistic CAD tools. We now provide an example of such a dichotomy with the issue of textures.

2.3 Textures

Engineering CAD and artistic CAD tools require an appropriate rendering of the 3D-designed product inside its environment, but for different purposes. Engineers require good rendering engines and libraries of materials to assess the object. Engineering CAD tools integrate a library of materials with properties as close to the real materials as possible. (Choi and Cheung 2005) proposed *upgrading the fit between the model and the real product, once manufactured*. Engineering CAD tools integrate libraries of *industrial materials* with properties such as color, stiffness, and weight, and take into account these properties for the entire object, for example, the effect of surface material stiffness and surface material weight on the entire stiffness and weight.

3D artists need to assess the future aesthetic properties of the future virtual product inside its virtual environment. (Lasram et al. 2012) proposed a new algorithm to simultaneously optimize the quality and rendering speed for procedural textures, and for their control and parameterization. Artistic CAD tools integrate libraries of natural appearances, such as skin, trees, fur, and sketch strokes, in addition to libraries of “virtual” appearances in areas such as fantasy, science fiction, and cartoons.

Thus, we can characterize two contrasting CAD families. Both have a requirement to “design a new artifact in a digital environment,” but have the same apparent issues regarding the computing engine and MMI in a similar way to any software, such as statistics software, office software, or enterprise resource planning. The literature helps to characterize some differences. The action logic is different: it is *modeling* in the case of engineering CAD and designing one *single artifact* in the case of 3D artist CAD. Hence, the issues with the computing engine are different: they are to increase model fidelity in engineering CAD and increase singularity and details in 3D artist CAD. Additionally, the MMI issues are also different: in engineering CAD, a MMI tends to increase the number of available features and combinations and the capacity to fit into multiple environments, such as a factory, logistics, retail, and with end-users, whereas in 3D

artist CAD, MMI attempts to enrich the virtual environment, the artifact rendering, and the fit between the virtual environment and the artifact itself.

2.4 CAD tools for ID

The questions are why and how ID CAD should be different and not simply a personalization or hybrid of the two types we have already identified. (Tovey 1997) explained the precise differences between the needs of IDs and their engineer colleagues. According to Tovey, IDs have the dual need to have tools for working on the aesthetic properties of products and their meaning, and to integrate with industrial firms and therefore work with 3D models intended for use in manufacturing objects, in the same way as engineers. (Fuge et al. 2012) confirmed this dual need. On the issue of “shape design”, IDs require a tool that has “to provide designers with the maximum flexibility in exploring a variety of shapes” and simultaneously to provide a “representation of the underlying surface. The authors finally proposed a direction for a renewed family of tools based on sketching and clay modeling techniques transposed into CAD environments (Tovey 2002).

Similarly, (Bae et al. 2008) proposed I Love Sketch software, which encourages IDs to express themselves with glyphs and pen–paper metaphors. This project is an attempt to introduce within a virtual environment an imitation of the traditional sketching process. (Thurgood and Clark 2001) used immersive interfaces to create services close to clay modeling, similar to those used by 3D artists. However, transferring from a CAD model to another type of data requires large computational efforts and breaking the link to the source file, where the design tree and all the design features are stored, to obtain what is called a “dead skin.” Another model conversion will be required if the designed prototype is validated. Thus, adding “free-form” modules to engineering CAD does not solve the issue of the compatibility with engineering CAD logic.

Finally, the literature on ID CAD tools underlines the mismatch between the needs of IDs and the digital tools they can access. Even with tuned-up engineering environments with specific interfaces, IDs still lack the capacity to express themselves freely because of the specific modeler kernels. Although creative prototypes have been proposed, the models they generate can be integrated into industrial environments only after data conversions (Thurgood and Clark 2001) that involve heavy losses of information. These data conversions are called the “*design-gap*” (Tovey 2002 ; Dorta et al. 2008). After more than 15 years of development, very few of these tools are used by IDs on a daily basis.

The challenge of making simple modifications to existing families of CAD tools to meet the needs of IDs appears impossible to solve. IDs lack dedicated tools that can address their specific reasoning and design processes while integrating them into industrial environments. Over the 15 years that have passed since the first papers on the use of CAD in industrial design, IDs are still struggling inside companies to find the right tools, to express themselves, and to communicate with other designers and fully participate in the entire design process.

2.5 Research question

This literature review helps to reject two common but too-simple hypotheses regarding ID CAD tools. CAD tools for IDs cannot be adapted from engineering CAD tools and CAD tools for IDs cannot be adapted from 3D artist CAD tools.

This results in the **research question: what is the specific design logic of IDCAD?** As seen with engineering CAD and 3D artist CAD, CAD software is coherent with the design action logic of the user.² One uses CAD to create “a new artifact in a digital environment,” and the literature helped us to clarify the meaning of “new” and “artifact” in each case. “New” for engineering CAD means a new combination of known features, whereas “new” for 3D artist CAD means an original single digital object. In engineering CAD, the user designs a model of the future product, whereas in 3D artist CAD, the user directly designs the final digital object. In IDCAD, it remains for us to determine the meaning of “artifact” and “new.”

3. Method

3.1 Analytical framework

To differentiate between the families of CAD software, we rely on recent findings in design theory that allow a better characterization of the specificities of design processes and the CAD tools mobilized for them. Recent advances in design theory help to characterize an ontology of design (Hatchuel et al. 2013 ; Hatchuel et al. 2011a):

- (1) Design involves managing the unknown. In design, there is a language of the unknown (Le Masson and Weil 2013). Design is a situated activity that implies the manipulation of knowledge about the known and concepts, which are the management of the unknown (Hatchuel and Weil 2009).
- (2) Design is a *generative process*. It allows the progressive creation of new objects, values, products, uses, or technologies, and the creation of new knowledge, which may or may not be related to the final artifact.
- (3) Design theories offer a *characterization of the evolution between the known and the unknown*. From the known, it is possible to organize the unknown, and the explorations of the unknown can generate new knowledge.
- (4) The performance of the design is, very generally, 2D: generativity of the design (is it new, original, etc.) and robustness (feasibility, domain of validity, etc.).

These results help us to build an analytical framework for CAD. CAD is a media to help move from one known–unknown estate to another through a generative process. We need a way to characterize the known and unknown. Many theories provide models for the structure of the known and unknown in design (Le Masson and Weil 2013 ; Hatchuel et al. 2011a). For this paper, we rely on classical language: DPs, FRs, and their relationships. This language, or a variant of it, can be found in many formal theories, such as general design theory (Yoshikawa 1981), axiomatic design (Suh 1990), and the coupled design process (Braha and Reich 2003). Zeng and Gu showed that this language of DPs and FRs could be useful for describing the design process and evolution of the artifacts in the design process (Zeng and Gu 1999a, b). DPs are the means of action of the designers; that is, these are what they can choose or parameterize, and where degrees of freedom exist. For example, DPs can be a form factor, dimension, material, or specific painting technique. FRs are the design targets of the designer. FRs are used to describe the

²This is even if engineering CAD or 3D artist CAD tools do not cover all facets of the action. For instance, engineering CAD tools support conceptual design quite poorly. In fact, future engineering CAD might try to address these design steps (see for instance Goel et al. 2012, Zeng and Horváth 2012).

expected performance of the design. They can be a tolerance, mass, dimension, specific meaning, or style.

The design process is characterized by the progressive exploration of the unknown and the expansion of the known. In FR-DP language, for instance, at the beginning of the design process, some FRs can be known, but how to obtain it is unknown, even if some potential DPs are known. Hence, the relationship between FRs and DPs is partly unknown.³ In the end of the process, one knows how to obtain the FRs with the known DPs. Moreover, often the relationship between FRs and DPs might take the form of a rule that has a certain domain of validity, so that at the end of the design process, one can actually design the entire object in the validity domain.

This simple framework helps us to characterize the design process steps, according to Table 1.

Table 1. Framework to analyze a CAD design process.

Category of CAD tool	Inputs	Outputs	Criteria of success of the considered design step
	DP: known/unknown FR: known/unknown Relationships between DP and FR: known/unknown	DP: known/unknown FR: known/unknown Relationships between them: known/unknown	

Note: For each CAD tool, one characterizes knowns and unknowns at an initial stage and knowns and unknowns and the final stage, and characterizes the type of performance that is targeted during the process.

The language of DPs, FRs, and their relationships helps us to characterize many design situations. For instance: (1) in engineering design, at the starting point, FRs are provided, but not the DPs to achieve them; (2) in industrial design, a brief is provided as a starting point and is a fuzzy FR (e.g., “a smart shopping cart,” see for instance Hatchuel et al. 2011b), and the design process progressively enriches the FR, for example, designing a new FR that can be associated to a smart shopping cart; and (3) in machine design, there are often many FRs given by the customer and many DPs to be used to address these FR (available processes, available material, etc.), but the relationship between the given FRs and the given DPs is partly unknown.

Our framework helps us to address our research question by characterizing the design situations to which each CAD tool is adapted. We proceed in two phases:

- (1) We first conduct an experimental comparison to characterize the design logic of CAD tools.
- (2) We then show that this design logic helps us to explain general trends in the development of CAD tools.

In the following, we provide detailed methods for each phase.

3.2 Case study

³In some cases, such as adaptive/variant design, the designers already know a large part of the relationship between DPs and FRs and they use CAD to fix the final details of this relationship.

Our experiment is based on the design of a new motorbike tank. The general brief was to design a “dynamic, muscular” tank. This was then evolved into three design briefs, each of which was adapted to be managed by the professional using particular CAD software. This product was chosen because it could be designed by any of the three categories of professionals. It exhibits technical, style, use, and aesthetic constraints.

This design exercise was applied to three specific CAD tools that are considered archetypal by designers and used by professionals on a daily basis:

- (1) The use case for the engineering CAD tool was implemented using SOLIDWORKS software. This CAD tool was developed for engineers working in various industries, such as robotics and musical instruments. It is based on the paradigm of procedural modeling. It is connected with many “workshops” for design analysis, such as resistance, weight measurement, and mechanical animation.
- (2) The use case for the artistic CAD tool was implemented using ZBrush software. This CAD tool was developed for 3D artists working in entertainment. It is specialized for the precise modeling of biological materials and original objects for virtual environments. It is based on the paradigm of clay modeling and connected with many “workshops,” such as rigging for animation and deformations, and rendering.
- (3) The use case for the ID CAD tool was implemented using CATIA Imagine & Shape software. This workshop was developed, with IDs involvement, for IDs working on the form and aesthetics of products in several industries. It is specialized for the modeling of qualitative shapes and draws from the entire CATIA environment. It is connected with many “workshops” suited for every step of the design process and most of the types of technical expertise (e.g., mechanics, electrics, and hydraulics), from ideation to design refinement and production. As mentioned above, very few CAD tools meet IDs’ expectations; however, previous experiments have shown that the CATIA Imagine & Shape kernel, interface, and integration have specific capacities that make it among the very few CAD tools that meet at least some IDs’ expectations (Arrighi et al. 2015).

Regarding the professionals, we had to avoid bias resulting from the user’s personality and education. Our issue was the comparison of the CAD tools and not the comparison of the professional users themselves. To avoid this type of bias and to focus on the comparison issue, we had to control the designer using the CAD software. Ideally, we needed the “same” professional user (same personality, tastes, etc.), but someone who was capable of using engineering, 3D graphic, and ID CADs with comparable and high capacity. We had the opportunity to rely on one such “ideal” participant. We identified a professional designer that was educated in all three professions and capable of using all three software systems. Thus, in this experimental study, we could compare the different software systems and minimize the bias introduced by different users. Note that there is no statistical relevance in the comparison, but analytically, we ensure that the differences we see are a result of different software systems used by this particular user and not the result of different users. We could not identify all the differences between the CAD software systems, but the differences we do identify can be attributed to those systems.

Because the same user conducted the three exercises, there might be a form of learning involved. This form of learning was maximized: the user performed the tasks in parallel, beginning with one CAD software system, then moving to another, and returning to the initial system. In fact, the designer worked in this way to be better able to determine the differences

between the three software systems. The experiment enabled the designer to characterize the differences between all three systems.

During the process and at the end, the designer was interviewed. The designer contributed to the comparison and differentiation of the three processes and three results that followed the analytical framework in Table 1.

3.3 Historical perspective

In the first phase, we characterized the design logic of CAD tools. This result was obtained with a specific use case and software systems. To generalize this result, in a second step, we validated our findings with a study of the historical developments of the CAD software systems. Relying on specific milestones in the development of the precursors of the cutting-edge contemporary software tools, we tracked how these specific categories of tools have evolved over time.

We conducted a historical analysis on each of the three software families: engineering design tools; graphic art tools, which included, for these broad categories, tools from 2D graphic creation to 3D animation tools and technologies; and industrial design tools.

4. Results

4.1 Design logic of CAD tools for engineers, 3D artists, and IDs

4.1.1 Designing with an engineering CAD tool

In the case of designing a “dynamic, muscular tank” using engineering CAD, the designer could not begin with the general brief. First, he had to translate it into known FRs. He explained that he considered that the tank had to achieve a specific volume, weight, and stiffness properties. He also took into account that the motorcycle tank would be manufactured and assembled; thus, it would have to encounter real-world constraints, such as rain, sun, and gravity, and would have to be compatible with norms, rules, and legal obligations. Regarding DPs, the designer relied on a set of materials and manufacturing processes that are available in the engineering CAD library. Thus, *before using CAD*, the DPs and FRs were fixed.

During the design process, the engineer configured a **pre-established and mandatory** set of DPs to achieve known FRs. The engineer used commands and functions step by step to construct the virtual mock-up, and by doing this generated what is known as a **design tree**. The design tree contained the functions and parameters, and the relationships among the different geometric entities. For instance, the designer specified certain dimensions of the tank to meet the volume criteria. The design tree also specified certain thickness parameters, for example, the position of ribs, to ensure the stiffness criteria (see Figure 1). *A single product was not designed; instead, a parameterized and reconfigurable base of rules were produced that could generate a family of products. The result of the design in engineering CAD was not a single product but rather an algebra of rules.* This algebra of rules defines how the components are spatially positioned, in addition to the links between them, the functions, and their parameters. This algebra can be mapped to an infinity of solutions. The engineer guaranteed that the set of created rules was robust to variations in DPs (e.g., different materials or manufacturing methods) or even FRs (e.g., modified volume requirements or a new legal regulation created during the design

process). Clearly, the designer could use irrelevant parameters so that the CAD tool allows parametric variations, but these variations are poorly related to important topological variations. Digital engineering appears to be a design process of **generalization**. The artifact is one representation of a “genre,” that is, it helps to address “generic variety.” The product is robust to changes in its DPs (components, dimensions, etc.) and its FRs (environment, etc.) – one simple illustration of this is the logic of tolerances in engineering design.

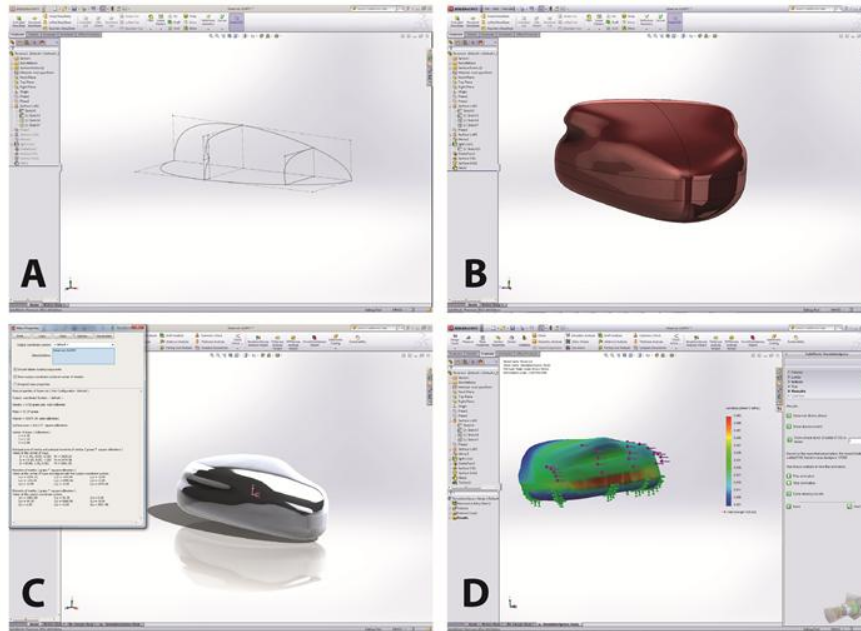


Figure 1. Four design steps of the engineering CAD tool, with finite element analysis (FEA) capability included: (a) sketching on 2D planes with a non-uniform rational basis spline; (b) general aspect of the final result; (c) stiffness simulation inside the tool, with FEA capability of the software system; and (d) volume and mass assessment inside the tool.

4.1.2 Designing with an artistic CAD tool

Using 3D artist CAD, the designer began with a brief for a “dynamic, muscular tank”. The designer noted that the design brief for the artistic 3D profession could also be a 2D sketch or illustration. The initial sketch provided indications of the environment and the object to be designed—the motorbike on which the tank would be installed, the environment into which the motorbike would be inserted, and so on. The designer then *extended* these elements so the artifact created was unique. The motorbike would never be manufactured; it stayed inside a virtual, controlled environment.

To obtain a muscular tank, the designer exaggerated the muscular aspect of his design; he added extra refinements and features. To address the FR “dynamic and strong,” he thought of a 5-year-old motorcycle that was used in the desert, and generated a specific DP: “there are impacts on the front lens, the paint is worn off and decolorized.” This motorbike was made specifically for a video game and driven by a specific character in a special environment, under specific lighting, and with a detailed paint job, including the detail of a specific little scratch on the side caused by a collision. This list of FRs was addressed by a combination of DPs: the driver, joints between the surfaces, number on the side, and color of the paint.

At the beginning, the designer then had a collection of DPs and FRs. He freely manipulated the DPs and FRs, adding some and suppressing others, under the guidance of an artistic director. Unlike his approach to the engineering experiment, he did not generate a generic relationship between DPs and FRs with a large validity domain. He **created a unique correspondence between a list of new DPs and new FRs**, with the longest list possible that included many “details.” Each detail had to convey meaning and intention to animate the virtual object.

Digital artistic design is a design process of **congruence** with the initial list, that is, the verbal brief or 2D sketched example, and **extension**, which adds a profusion of details to obtain a unique artifact, that is, to achieve a form of singularity.



Figure 2. Six design steps of the 3D artist CAD tool: (a) muscle reference, (b) import a low resolution primitive, (c) raw deformation of the low definition primitive, (d) sculpting the ribs at low definition, (e) sculpting additional ribs at high definition, and (f) final result with chrome material applied.

4.1.3 Designing with an ID CAD tool

Using the ID CAD tool, the designer began with a “dynamic and muscular tank,” but he also added other elements, such as global dimensions, the process of fabrication, and manufacturing cost. The designer began with references, such as sketches, pictures, or photos. Similar to the engineering CAD experiment, the designer took into account that the tank would have to encounter real-world constraints, such as rain, sun, and gravity, and that it must be compatible with norms, rules, and legal obligations. Simultaneously, it had to satisfy specific usage and aesthetic constraints that were difficult to measure.

The design process required the creation of details. For the first step, the designer explored a high density of details in a similar way to a 3D artist working with CAD tools. For the second step, the designer omitted some details to fit into a small set of parameters in the data

format of a manufacturable file, thereby creating a conceptual model for the tank very similar to the type of files produced by engineering CAD.

This conceptual model synthesized and simplified a large variety of DPs, such as points, lines, curves, surfaces, and their parameters, into a smaller set that was compatible with the constraints of engineering design and manufacturing. This limited set was intended to convey the dynamism and muscularity of the tank. In our case (see Figure 3), the conceptual model is the instruction about the character lines of the tank that make it dynamic and muscular, in addition to how the character lines respect the surface quality: their position, thickness, orientation, style, and artistic influences. Software requires the ability to *extend the list of DPs and FRs and synthesize the explorations into a conceptual model*.

In the analytical framework (see Table 2), the design process generates new DPs and FRs and *it also generates new relationships between these DPs and FRs. This process creates a new algebra* that can be activated and parameterized on demand once characterized.

In the case of ID CAD, the result is not only the particular shape obtained but *also the conceptual model that underlies this particular shape*, that is, it helps to create this shape but also a family of shapes sharing a common character. Hence, it possesses some of the properties of engineering CAD (i.e., model and genericity) and some of artistic CAD (i.e., generation of FRs and DPs, and singularity). **The capacity to create conceptual models makes this type of tool suited for what we call generic singularity.** The artifact designed by the ID is a specific configuration of his conceptual model.

There is the question of what is “new” in this process. Similar to his role as a 3D artist, when working as an ID, the designer used CAD to express himself freely to create an original artifact. But he also had to take into account the constraints of the development of the real object (development, production, retail, marketing, etc.). Newness is then defined as the capacity to make an original model and not only a single digital tank under the industrial constraints. Ideally, this model would be valid not only for the tank of the motorbike, but also it might be a model of “muscularity” that would be valid for a large set of objects, for example, the tank, wheels, helmet, and motorcyclist's jacket. The performance of the designer using ID CAD creates a conceptual model that is valid on the largest domain possible.

Finally, there is the question of what comprises a conceptual model that is new and valid for a large set of objects. This corresponds to what is usually called a “style,” for example, the style of an artist or time period. We finally determine a deep and critical identity feature of an ID: an ID creates a personal style or the style of the age. Additionally, this style corresponds to a new but stabilized, constant relationship between new or old DPs and new or old FRs. While the 3D artist creates a singular object with CAD and the engineer creates a generic variety of objects with CAD, an ID creates a **generic singularity by creating a new style**, which is a new conceptual model.

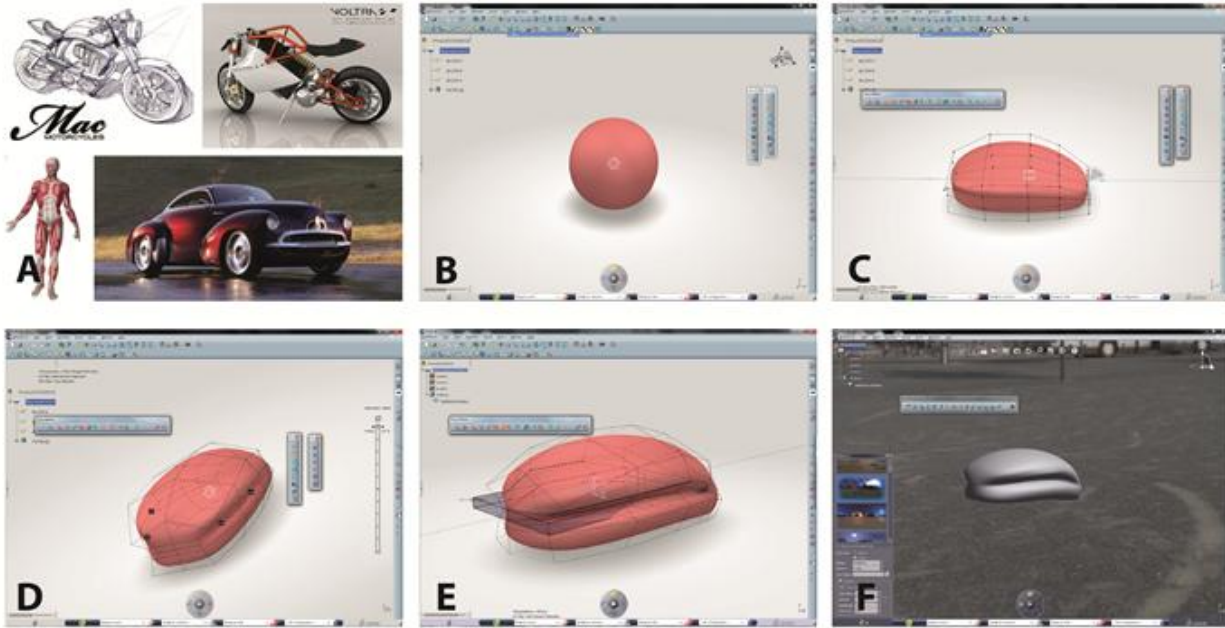


Figure 3. Six design steps of the ID CAD tools: (a) reference, (b) importing a primitive, (c) playing with the shape manipulators, (d) applying constraints over the manipulation cage, (e) editing the parameters of the rib, (f) final result inside an urban environment.

4.1.4 Conclusions on the archetypal CAD design tools

Table 2. Characterization of the three archetypal CAD tools.

Category of CAD tool	DP-input FR-input and relationships between them	DP-output FR-output and relationships between them	Criteria of success of the considered design step
Engineering (SOLIDWORKS)	<ul style="list-style-type: none"> • Mandatory DP-input • Mandatory FR-input • Partial relationships between known DP and FR 	<ul style="list-style-type: none"> • DP-output = DP-input • FR-output = FR-input • Robust domain of solutions linking FR and DP 	<ul style="list-style-type: none"> • Size of the domain • Degree of performance of the local optimums • Robustness of the domain to both DP and FR modifications
Artistic (ZBrush)	<ul style="list-style-type: none"> • List of desired DP-input • List of desired FR-input • List of imbricated DP and FR that describe a unique prescription 	<ul style="list-style-type: none"> • New list of DPs, some similar to DP-input, others different • New list of FRs, some similar to FR-input, others different • New list of imbricated DP and FR with more details and meaning 	<ul style="list-style-type: none"> • Extension of the list • Density of details • Unicity of the list • Superfluous DP responds to superfluous FR giving the next designer the most unique intention • Congruence between initial and final list
Archetypal industrial design (CATIA Imagine)	<ul style="list-style-type: none"> • List of desired or mandatory DP-input • List of desired or 	<ul style="list-style-type: none"> • New list of FR, including mandatory FR 	<ul style="list-style-type: none"> • Fabrication of the conceptual model • Robustness of the conceptual

& Shape)	mandatory FR-input <ul style="list-style-type: none"> • Partial relationships between known DP and FR • List of imbricated DP and FR that describe a unique prescription 	<ul style="list-style-type: none"> • New list of DP, including mandatory DP • New conceptual model 	model to both DP and FR modifications <ul style="list-style-type: none"> • Degree of performance and originality of the optimal conceptual model configuration • Congruence between initial prescription and final model
----------	--	--	--

After completing the analysis based on the use cases of archetypal tools, we can characterize, from a design perspective, the differences among the categories of CAD tools and why they are suited to their specific design users.

- Engineering CAD tools seek to generate the best possible algebra of rules between the *prescribed* FRs and DPs. This algebra must be valid on a domain of DPs and FRs; the wider and more robust this domain, the better. Thus, there is a logic of *genericity* in engineering CAD.
- Artistic CAD tools seek to generate the most detailed, refined, and unique list possible of coherent FRs and DPs. This list must be as long and as original as possible, in congruence with the prescription. There is no modeling logic; there is a logic of *singularity*.
- ID CAD tools can work with an incomplete set of DPs and FRs; they help to extend the list, creating a sense of “muscularity” and DPs that can be associated with “muscularity” and they help create the associated conceptual model by linking DPs and FRs that are valid on a large validity domain, which can be referred to as a new **style**. There is a logic of *generic singularity*.

4.2 Historical evolution of CAD tools

We now verify whether this hypothesis is correct from a much wider perspective. There are many works on the history of CAD tools (Arrighi and Kazakçi 2013) (Fridenson 2015). We rely only on some main milestones in each profession.

4.2.1 Evolution of engineering practices

We summarize the history of engineering CAD in Table 3. Engineering CAD was a technique to help to relate known FRs and known DPs, such as shape and material. The evolution is marked by the need to *increase* the list of available DPs and FRs, and the capacity to relate complex sets of DPs and FRs.

Table 3. Historical evolution of engineering CAD from the design theory point of view.

Types of engineering CAD	DPs	FRs	Issues
CAD/CAM (1970s, early 1980s)	2D-draughting Wire-frame and surface modeling Deformation modeling (Bezier logic, see	Specific shapes to meet product and process requirements	Capacity to automatically redesign Data compatible with computerized numerical

	Fridenson 2015)		controlmachines
Solid modeling (late 1980s)	Parametric models Associative model Feature-based model	Manufacturability Composition Tolerances	
Digital mock-up (1990s)	Multiple engineering sciences and skills	Multiple engineering sciences and skills	Mix multiple engineering languages
Product life cycle (2000s)	Process assembly After-sales Maintenance Retail	Process assembly Maintenance After-sales Retail	Integrate new languages

4.2.2 Evolution of 3D artist practices

The history of 3D artist CAD shows how 3D graphic CAD has been progressively open to be able to integrate denser details and new “types” of details. We summarize the steps in Table 4.

Table 4. Historical evolution of 3D graphic CAD from the design theory point of view.

Types of engineering CAD	DPs	FRs
Wireframe (1960s)	Wireframe	Only contours
Polygons (1960s)	Polygons: deformable polygons, calculates reflection angles	Surfaces; deformable, animated surfaces, Lighting
Tessellation (1970s)	Tessellation and skinning	Texture
Subdivision (1980s)	Subdivision techniques: uniform basis-spline curves from artist’s work	Smooth surface rendering
“Clay modeling” (1990s–2000s)	Voxel	Rich details on organic surfaces

4.2.3 Evolution of ID CAD tools

For some time, there were very few dedicated tools for IDs. Despite this, the following challenges have progressively been overcome by some software systems for IDs:

- translating between the different data formats used by engineering design tools and industrial design tools;
- representing objects with a very high surface quality, which is required especially to achieve high levels of precision in manufacturing; and
- maintaining both a holistic view of the object being designed and the ability to work in very local detail. In practice, this implies being able to realize modifications that need to be propagated only locally.

- Finally, ID CAD tools support the creation of conceptual models (i.e., ID styles) in two ways:
- (1) ID CAD tends to hybridize engineering CAD logic and 3D artist CAD in a two-step process: the designer first uses 3D artist CAD to have great freedom of exploration, which is possible with artistic CAD tools. For the second step, the ID uses engineering CAD to apply constraints. Some current software prototypes allow the designer to explore every dimension while designing and then the user activates constraints, such as the surface quality or process of manufacturing.
 - (2) ID CAD provide a specific environment where IDs are aware of the constraints and are bound to make only models of a given surface quality, Hence with ID CAD, IDs can concentrate on the generation of the conceptual model and its originality because they have the guarantee that his model will fit the industrial manufacturing constraint.

This corresponds to two strategies to generate a style with IDCAD:

Strategy 1: maximal extension, then generation of a conceptual model.

Strategy 2: modeling inside the constrained environment. The conceptual model is created by the embedded constraint inside the tool.

4.2.4 Conclusions on the historical study of CAD design tools

This historical analysis supports the previous findings (see Table 5):

- (1) Engineering CAD relies on genericity: designers design a product, for mass production or one-offs, that have to fit into complex, real external conditions, such as users' behavior, robustness to external events, and conditions, and the trend consists of widening the set of rules to be managed with CAD.
- (2) For 3D artist CAD tools, the DP are mainly graphic and the issue is to obtain denser and more precise shapes while addressing all the facets of an object in its environment to convey the meaning of the FR. CAD tools help to design and control as many dimensions as possible to obtain a unique and original design.
- (3) IDs, with both technical and artistic competencies, have been trapped between these two worlds until very recently. New ID CAD tools help at the present time to address a large variety of FRs inside industrial environments and to have the capacity to generate new conceptual models, that is, to act in a **"style making" logic**. They combine the singularity of personal style with the genericity of a style applied to many different artifacts.

Table 5. Characterization of the three archetypal CAD tools.

Category of CAD tool	Remarkable evolutions
Engineering	Variety and number of DP and FR Complexity of the base of rules and their robustness
Artistic	Variety and number of DP and FR Density and unicity of the combination of DP and FR Capacity to add an infinite number of relevant details when needed
Industrial designer	Manage the "style making" logic, such as compatibility and rules

	Extend the notion of style to other dimensions, (e.g., use scenario and emotion)
--	--

5. Discussion and conclusion

5.1 Characterization of CAD tools

We have characterized the underlying design processes of the CAD tools for three professions using our design theory model.

5.1.1 Engineering CAD tools

For engineering CAD tools, the design expansion is the making of a rule defining the relationship between DPs and FRs. The target for this class of designers is to ensure that the design fits within a variety of external conditions, that is, it will be generic for these conditions. The designers create a set of rules and not one single object. The global tendency of this type of software is the capacity to generate rules addressing more types of DPs and FRs, which implies more heterogeneous disciplines, and to validate them across wider and more precise domains. We call this tendency **generic variety**.

5.1.2 CAD tools for 3D artists

3D artists require the ability to expand the list of DPs and FRs. There are no rules and domains, only a unique result that must be as dense as possible in detail and meaning. The target is a singular and original realization. The global tendency of this type of software is to make more FRs achievable as a result of an increase in the number of DPs available. We call this tendency **maximal singularity**.

5.1.3 CAD tools for ID

IDs seek new and original DPs and FRs, and simultaneously try to generate new rules between those DPs and FRs. This process is different from engineering design because the rule is not the unique target: the DP and FR must also be formulated and generated. It is also different from 3D artist CAD tool design because the expansion of DPs and FRs is not sufficient: there is a need to converge on a stabilized relationship between DPs and FRs on a large validity domain. The target is to obtain a generic singularity, which is generic because the new rule is suitable on a validity domain and a singularity because it relies on new DPs and new FRs; it is the design of a new **conceptual model, or a new style. ID uses CAD to obtain generic singularity.**

5.2 Perspectives

This work helps to explain CAD and, more specifically, CAD for ID, by relying on advances in design theory.

- (1) It helps to clarify what is designed with CAD and to which of the design situations each is adapted.

- (2) It shows that ID CAD has to be coherent with ID design logic, not the creation of a single artifact, but the creation of “the style of their age,” as claimed by Bauhaus, one of the most famous design schools(Le Masson et al. 2015).
- (3) Moreover, it helps to clarify two IDCAD logics: expansion and then selection, or expansion driven by constraint;that is,managing with constraints creatively.

More generally, it appears that IDCAD confronts the critical issue of *creating* a style in software. This issue appears as new and critical for CAD: engineering CAD uses existing rules, DPs, and FRs to instantiate generic models (modeling logic), and the 3D artist actually designs a single artifact. IDCAD invents dimensions and also ensures that their creation has a large validity domain, that is, that the new “style” can be applied to a large set of “objects.”

In further research, this effort to create models might be related to contemporary issues in other domains:

- Contemporary machine learning now extends beyond identifying known patterns and tends to use big data to discover unknown patterns (Kazakçi 2014). Hence, linking IDCAD and contemporary machine learning might be fruitful. Other areas of interest include IDCAD applied to big data issues or, conversely, machine learning algorithms applied to IDCAD.
- IDs are not the only professionals to create models. Creative scientists also try to extend beyond the validation of hypotheses and aim at creating new theories for new hypotheses. Thus, software that helps to create models might be relevant to them.

Thus, we have shown that the issue of IDCAD is far from being the orphan issue of one single profession; instead, it echoes the contemporary issues of science: invention of style and model creation.

References

- Arrighi P-A, Kazakçi AO (2013) Lack of integration between engineering industrial design processes. . In: *21st International Conference on Engineering Design, 19th, 22nd August 2013*, Seoul, Korea, 2013.
- Arrighi P-A, Le Masson P, Weil B (2015) Addressing constraints creatively: how new design software helps solve the dilemma of originality and feasibility". *Creativity and Innovation Management* 24 (2):247-260.
- Bae S-H, Balakrishnan R, Singh K (2008) ILoveSketch: As-natural-as-possible sketching system for creating 3D curve models. In: *ACM Symposium on User Interface Software and Technology*, Monterey, CA, October 19-22, 2008 2008.
- Bettig B, Shah J (2001) Derivation of a standard set of geometric constraints for parametric modeling and data exchange. *Computer-Aided Design* 33 (1):17-33.
- Bodein Y, Rose B, Caillaud E (2012) Decisional model for KBE implementation in a commercial CAD software. *Computer Aided Design Applications* 9:121-131.
- Braha D, Reich Y (2003) Topological structures for modelling engineering design processes. *Research in Engineering Design* 14 (4):185-199.
- Brunetti G, Golob B (2000) A feature-based approach towards an integrated product model including conceptual design information. *Computer-Aided Design* 32 (14):877-887.
- Chen X, Gao S, Yang Y, Zhang S (2012) Multi-level assembly model for top-down design of mechanical products. *Computer-Aided Design* 44 (10):1033-1048.

- Choi SH, Cheung HH (2005) A multi-material virtual prototyping system. *Computer Aided Design* 37:123-136.
- Deschênes F, Ziou D, Fuchs P (2004) An unified approach for a simultaneous and cooperative estimation of defocus blur and spatial shifts. *Image Visual Computing* 22:35-57.
- Dorta T, Perez E, Lesage A (2008) The ideation gap: hybrid tools, design flow and practice. *Design Studies* 29:121-141.
- Dyn N, Floater MS, Hormann K (2009) Four-point curve subdivision based on iterated chordal and centripetal parameterizations. *Computer Aided Geometric Design* 26:279-286.
- Fridenson P (2015) Inventeur salarié, automatisation et entreprise : expériences internationales, tensions et dynamiques. In: Michel A, Benoit S (eds) *Le monde du génie industriel au XXe siècle : autour de Pierre Bézier et des machines-outils*. Pôle éditorial de l'UTBM, Belfort,
- Fuge M, Yumer ME, Orbay G, Kara LB (2012) Conceptual design and modification of freeform surfaces using dual shape representations in augmented reality environments. *Computer-Aided Design* 44 (10):1020-1032.
- Goel AK, Vattam S, Wiltgen B, Helms M (2012) Cognitive, collaborative, conceptual and creative — Four characteristics of the next generation of knowledge-based CAD systems: A study in biologically inspired design. *Computer-Aided Design* 44 (10):879-900.
- Gross MD, Do E (1996) Ambiguous intentions: a paper-like interface for creative design. Paper presented at the *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, Seattle, WA, pp 183-192,
- Hatchuel A, Le Masson P, Reich Y, Weil B (2011a) A systematic approach of design theories using generativeness and robustness. In: *International Conference on Engineering Design, ICED'11*, Copenhagen, Technical University of Denmark, 2011a. p 12
- Hatchuel A, Le Masson P, Weil B (2011b) Teaching Innovative Design Reasoning: How C-K Theory Can Help to Overcome Fixation Effect. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 25 (1):77-92.
- Hatchuel A, Weil B (2009) C-K design theory: an advanced formulation. *Research in Engineering Design* 19 (4):181-192.
- Hatchuel A, Weil B, Le Masson P (2013) Towards an ontology of design: lessons from C-K Design theory and Forcing. *Research in Engineering Design* 24 (2):147-163.
- Igarashi T, Matsuoka S, Tanaka H (2007) Teddy: a sketching interface for 3D freeform design. Paper presented at the *ACM SIGGRAPH 2007*, New York, NY,
- Kang Y, Kim H, Suzuki H, Han S (2013) Feature-based 3D CAD modeling on smart device using multi-touch gesture. *International Journal of CAD/CAM* 13: 49-62.
- Kazakçi AO (2014) Conceptive artificial intelligence: insights from design theory. Paper presented at the *International Design Conference Design 2014*, Dubrovnik, Croatia,
- Keefe D, Feliz D, Moscovich T, Laidlaw D, LaViola JJJ (2001) CavePainting: a fully immersive 3D artistic medium and interactive experience. Paper presented at the *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, ACM, New York,
- Kelly JC, Maheut P, Papalambros PY, Petiot JF (2011) Incorporating user shape preference in engineering design optimisation. *Journal of Engineering Design* 22:627-650.
- Kowalski MA, Markosian L, Northrup JD, Bourdev L, Barzel R, Holden LS, Hughes JF (1999) Art-based rendering of fur, grass, and trees. Paper presented at the *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*,

- Kwok T-H, Wang CCL (2014) Shape optimization for human-centric products with standardized components. *Computer-Aided Design* 52:40-50.
- Lasram A, Lefebvre S, Damez C (2012) Procedural texture preview. *Computer Graphic Forum* 31:413-420.
- Le Masson P, Dorst K, Subrahmanian E (2013) Design Theory: history, state of the arts and advancements. *Research in Engineering Design* 24 (2):97-103.
- Le Masson P, Hatchuel A, Weil B (2015) Design Theory at Bauhaus: teaching “splitting” knowledge. *Research in Engineering Design* (accepted).
- Le Masson P, Weil B (2013) Design theories as languages for the unknown: insights from the German roots of systematic design (1840-1960). *Research in Engineering Design* 24 (2):105-126.
- Liu H, Maekawa T, Patrikalakis NM, Sachs EM, Cho W (2004a) Methods for feature-based design of heterogeneous solids. *Computer-Aided Design* 36 (12):1141-1159.
- Liu X, Dodds G, McCartney J, Hinds BK (2004b) Virtual DesignWorks—designing 3D CAD models via haptic interaction. *Computer Aided Design Applications* 36:1129-1140.
- Liu Y, Ritchie JM, Lim T, Kosmadoudi Z, Sivanathan A, Sung RCW (2014) A fuzzy psycho-physiological approach to enable the understanding of an engineer’s affect status during CAD activities. *Computer-Aided Design* 54:19-38.
- Loop C, Schaefer S (2008) Approximating Catmull-Clark subdivision surfaces with bicubic patches. *ACM Transactions on Graphics* 27 (8).
- Loop C, Schaefer S, Ni T, Castano I (2009) Approximating subdivision surfaces with Gregory patches for hardware tessellation. *ACM Transactions on Graphics* 28:151.
- Mandil G, Serré P, Mireille M, Desrochers A (2011) Geometrical tools for the description and control of functional specifications at the conceptual design phase. *IJODIR* 5 (1).
- Mishkinis A, Gentil C, Lanquetin S, Sokolov D (2012) Approximate convex hull of affine iterated function system attractors. *Chaos Solitons Fractals* 45:1444-1451.
- Nealen A, Igarashi T, Sorkine O, Alexa M (2007) FiberMesh: designing freeform surfaces with 3D curves. *ACM Trans Graph* 26:41.
- Nguyen TA, Zeng Y (2014) A physiological study of relationship between designer’s mental effort and mental stress during conceptual design. *Computer-Aided Design* 54:3-18.
- Ouertani MZ, Baïna S, Gzara L, Morel G (2011) Traceability and management of dispersed product knowledge during design and manufacturing. *Computer Aided Design Applications* 43:546-562.
- Rosen D (1993) Feature-based design: Four hypotheses for future CAD systems. *Research in Engineering Design* 5 (3-4):125-139.
- Rosen D, Peters T (1993) Special issue: Advances in representation and reasoning for mechanical CAD. *Research in Engineering Design* 5 (3-4):123-124.
- Salehi V, McMahon C (2011) Development and Application of an Integrated Approach for Parametric Associative CAD Design in an Industrial Context. *Computer-Aided Design and Applications* 8 (2):225-236.
- Shah J, Rogers M (1993) Assembly modeling as an extension of feature-based design. *Research in Engineering Design* 5 (3-4):218-237.
- Stergiopoulos P, Fuchs P, Laurgeau C (2003) Design of a 2-finger hand exoskeleton for VR grasping simulation. Paper presented at the *Proceedings of Eurohaptics*, Dublin Ireland, July 2003
- Suh NP (1990) *Principles of Design*. Oxford University Press, New York

- Thurgood P, Clark J (2001) Use of 3D non-contact digitisation and 3D touch modelling systems within automotive seating design. Paper presented at the *Proceedings of TCT Conference* Chester, UK,
- Tovey M (1997) Styling and design: intuition and analysis in industrial design. *Design Studies* 18 (1):5-31.
- Tovey M (2002) Concept design CAD for the automotive industry. *Journal of Engineering Design* 13 (1):5-18.
- Whited B, Daniels E, Kaschalk M, Osborne P, Odermatt K (2012) Computer-assisted animation of line and paint in Disney's Paperman. Paper presented at the *SIGGRAPH Talks*
- Xu Z, Zhang J, Li Y, Jiang S, Sun Y (2013) Product modeling framework based on interaction feature pair. *Computer-Aided Design* 45 (12):1591-1603.
- Yoshikawa H (1981) General Design Theory and a CAD System. In: Sata T, Warman E (eds) *Man-Machine Communication in CAD/CAM, proceedings of the IFIP WG5.2-5.3 Working Conference 1980 (Tokyo)*. Amsterdam, North-Holland, pp 35-57
- Zeng Y, Gu P (1999a) A science-based approach to product design theory: Part 1: formulation and formalization of design process. *Robotics and Computer Integrated Manufacturing* 15:331-339.
- Zeng Y, Gu P (1999b) A science-based approach to product design theory: Part 2: formulation of design requirements and products. *Robotics and Computer Integrated Manufacturing* 15:341-352.
- Zeng Y, Horváth I (2012) Fundamentals of next generation CAD/E systems. *Computer-Aided Design* 44 (10):875-878.
- Zhang DH, Yan FX, Hou ZX, Kang WK (2009) Virtual clay modeling system with 6-DOF haptic feedback. *Material Science Forum* 628:155-160.