Chapter 2
Integrated Product Design and Development

This chapter discusses the evolution of product development approaches through time. It reviews the serial or sequential approach that was adopted by the companies under the influence of Industrial Revolution and Fordism. Then the Integrated Product Development (IPD) approach is presented and discussed. It's worth mentioning that IPD was influenced by the Computer Integrated Manufacturing (CIM) proposal that emerged in the 80s as an evolution of the Ford manufacturing system. IPD keeps the benefits from the former approach (shorten price, shorten time-to-market, augmented quality) while fixes its shortcoming such as reworks, lack of communication amongst technical areas etc. IPD prescribes the structuring of two main pillars, namely, multifunctional or IPD teams and DFX (Design for eXcellence) design tools. After presenting some practical examples of the usage of DFX design tools, this chapter introduces the novel concept of integrative design variables (IDV): there is a target value associate to them; they are affected and affect most of the design decisions and their meaning is easy to grasp. Cost, weight, center of gravity are IDV examples. The IPD concept goes far beyond standard products such as cars, aircrafts and washing machines. At the end of the chapter you'll find the IPD applied to academic or technical assessment.

2.1 Introduction

As discussed in the previous chapter, Product Design and Development can be seen as a process and, consequently can be modeled. Figure 2.1 depicts the simplest possible model for such a process.

Although simple, the model contains the main activities of the product development process (PDP), namely, needs identification, synthesis of the product, and evaluation of the design alternatives for the product.

The PDP ought to start with the customer needs identification. Sometimes these needs are presented in a broad way, such as the need for reducing atmosphere CO_2 emission from the airplane, and other times in a very strict sense, such as defining an automatic procedure for installing aircraft rivets.

M.V.P. Pessôa and L.G. Trabasso, *The Lean Product Design and Development Journey*, DOI 10.1007/978-3-319-46792-4_2

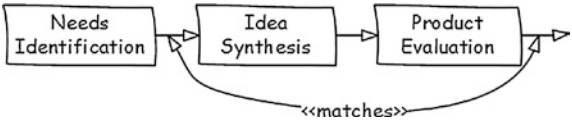


Fig. 2.1 The simplest model of product development

Some people think that PDP is all about synthesizing ideas and conceiving new products. Synthesis plays an important role within the product design and development, but is merely one activity of the PDP and has equal weight as the others.

The evaluation activity verifies whether the synthesized product meets the needs identified at the very beginning of the process. If not, a design loop is established up to the point that the needs are met. Evaluation of the product alternatives is as important as the needs identification and the product synthesis. How engineers have approached these activities through time has defined Product Design and Development evolution.

2.2 Sequential Product Design and Integrated Product Design

During the Industrial Revolution of the 18th century, a company that designed and built steam engines had argued that its engine was better than its counterpart. Naturally, the other company had the opposite opinion. Quarrels like this only ceased after the publication of the thermodynamics laws that were used as quantitative criteria for evaluating the best design alternative for the steam engine.

How do we evaluate design concepts? The first move for many of people is to base it upon the product functionality. Suppose you are given the following requirements (needs): design a product which is capable of lifting a 500 kg block of steel to a height of 1 m from the floor, moving it along a 3 m straight path at a speed of 0.5 m/s and lowering it back onto the floor. What would come to your mind? Steel cables, hooks, electric motors, brakes, axles, pulleys etc. Putting all the components together, your product would work, or should we say it would *function* because it resembles a mechanical hoist.

In this example, have you thought about the best way of assembling the components? How could it be easier for the maintenance personnel to execute their jobs? How should the components be manufactured in order to facilitate access to the tooling? These are questions beyond the functional evaluation which need answered during the product life cycle.

In the serial product development (SPD) (Fig. 2.2a), on the one hand, only the functionality of the product is taken into account during synthesis. Regarding the previous example, if the product designed does lift the 500 kg steel block, moves it, and puts it back on the floor, it does function! Manufacturing, assembly, maintenance issues and so forth are solved later by somebody else.

On the other hand, in the integrated product development (IPD) approach (Fig. 2.2b), the requirements from the product lifecycle areas such as: design,

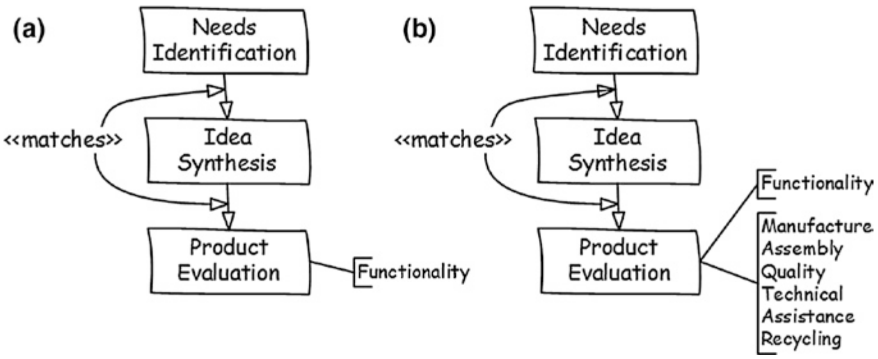


Fig. 2.2 Serial (**a**) and integrated (**b**) model of PDP

manufacturing, assembly, maintenance, disposal, and so on, are also considered, weighed, discussed, and balanced at the conceptual phase of the product development. [1] As a result, the outcome from integrated product development is a product which is designed not only to work, but also to be easily and cheaply manufactured, assembled, tested, maintained, and recycled.

By comparing the models of the serial product development and the integrated product development, one can realize that the latter encompasses the former. IPD expands the horizon of the product evaluation by trying to take into account all the technical areas and phases the product goes through during its lifecycle.

2.3 Serial Product Development

Before the Industrial Revolution, there existed the most–ever–integrated product development. Think of an artisanal shoemaker. He knew how to design the shoes and he mastered all necessary tools and tooling for manufacturing the shoes which would fulfill all the functions and needs from his neighborhood. The shoemaker knew the tastes of his customers and would ask from time to time whether a small repair was due. Market needs, product conceptual design, manufacturing, assembly, and maintenance were integrated in the shoemaker's head at the speed of synapses (Fig. 2.3).

With the Industrial Revolution also came the division of work into specific technical areas. The product development process mirrored the serial production line, thus adopting the serial approach as well.

One might righty argue the benefits brought by the industrial revolution: product costs reduction, production increase and a raise in quality standards. These benefits still exist today, but it stands to reason that the mass production era split the technical areas of the product development process into separate departments (i.e., silos), based upon highly skilled people within them, but with almost no interaction among them.

Fig. 2.3 Synergy of the integrated product development

Within the PDP, a typical manner of work by the multiple departments is to finish their jobs as quickly as possible and throw them over the “wall” to the next department. Suppose that the designer from Fig. 2.4 “threw” a blueprint in which a 3.3 mm diameter hole has been drawn. As soon as the blueprint lands in the manufacturing department, the technician will realize that there are no 3.3 mm commercially available drills. Naturally, the manufacturing technician could not choose an available drill whose diameter was close to that specified by the design department. Then the project oscillation begins: the manufacturing department writes down a design change request that will be analyzed and eventually implemented after some interaction loops.

The barrier metaphorically represented in Fig. 2.4 extends to all the areas participating in the PDP, creating a great challenge to integration.

Rescuing the integration of the PDP is therefore a challenge, but this should not hamper the attempt to achieve it. One can find the motivation to pursue it by looking at the negative consequences of the serial product development:

- Production is not considered in the conceptual phase of PDP but only at the very phase of production where the product modifications, if needed, are more difficult to implement and costly.
- Product data is fragmented so that each technical area has its own product data representation.
- Product development is driven by milestone dates associated to each development phase; thus, putting pressure on technical specifications and drawings release. As a consequence, few design alternatives are evaluated.

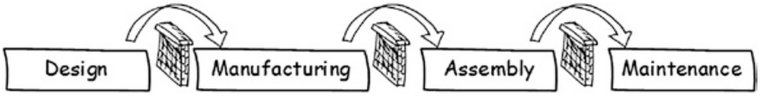


Fig. 2.4 Organizational barriers for the integrated product development approach

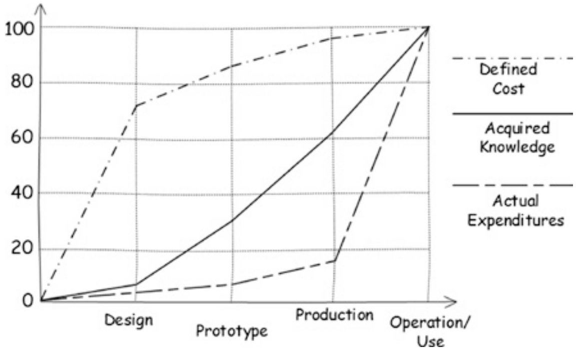


Fig. 2.5 Typical behavior of the product development process

These negatives consequences could be used for justifying the replacement of the serial to the integrated approach for product development. However, a stronger and more eloquent reason could be drawn from Fig. 2.5.

The PD life cycle phases are represented at the abscissa axis. The ordinate axis shows the percentage magnitudes of three important variables within PDP, namely, defined cost to implement a given PDP phase, knowledge acquired about the product, and actual incurred cost of the product defined.

Figure 2.5 draws attention to the 75 % mark of the defined cost regarding the conceptual (design) phase of PDP. This means that 75 % of the overall forthcoming cost of the product is defined at the conceptual phase of PDP. It is not difficult to figure out the causes: the designer has to define the shape, geometry, and features of a product which are strictly related to the manufacturing process. In addition, the designer ought to define, but not yet buy, the materials of the components and parts. To complete the product specification, the engineer/designer has to define geometric and dimensional tolerances of the components and parts as well as define the surface finish.

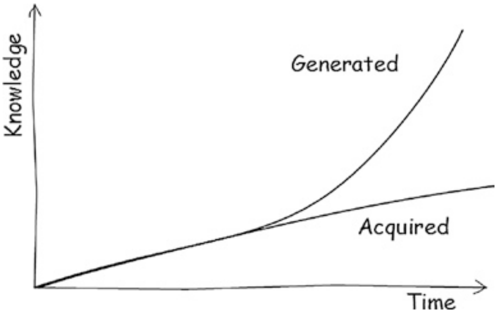
The gap between knowledge and cost decisions implies that many decisions are made based on wishful thinking, therefore causing rework and correction loop-backs during the remainder of the PDP and product life cycle.

2.4 Integrated Product Development: A Rescue Movement

Integrated product development is all about rescuing the interaction among the technical areas concerning the product in which requirements are taken into account and balanced for the benefit of the product. This is a “rescuing” movement because this integration once existed and was lost. However it is not possible to rescue the integration as it was at the artisanal production level. It is no longer possible for a single person to keep with all the information and have all the knowledge needed to consider all aspects of the lifecycle of a typical complex product of current times, as shown in Fig. 2.6. Even if we drop the complexity of

Fig. 2.6 A highly complex product: EMBRAER KC-390. *Source* Disclosure Embraer

Fig. 2.7 Knowledge production versus knowledge acquisition



the product to a cell phone, the challenge for recreating the integration environment remains.

Indeed, knowledge generation poses a great challenge for successfully rescuing the integration of the product development process. At this very moment, a great amount of brand-new knowledge is being produced all over the world so that it is literally impossible to catch up to such a pace as Fig. 2.7 represents.

Considering what was presented in Fig. 2.5, by comparing the knowledge and expenditures curves within the conceptual phase of PDP, one can realize that most of the budget commitment is taken with a low degree of knowledge. These decisions should take into account not only aspects from the conceptual phase, but from other phases, such as manufacturing and assembly, as well. Thus, the decisions and definitions of the conceptual phase ought to be taken with a higher degree of knowledge as depicted as in Fig. 2.8.

One of the objectives of IPD is to increase the knowledge of the product at the earliest phase of the PDP, and supporting the decisions that must be taken at this moment. The actual expenditures line indicates that investments should be made in

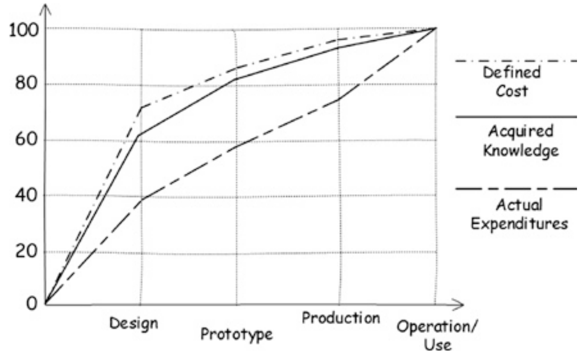


Fig. 2.8 A new proposal for PDP: knowledge build up through integration

order to create this knowledge. To gather information and requirements of dimensional and geometric tolerances, for instance, the company might contract and pay for consultancy in that field. It's worth mentioning that the percentage of 75 on the defined cost remains the same in the IPD scenario because the decisions about product geometry, materials, tolerances, and surface finishing need to be taken regardless of the increase in knowledge about the product.

If we analyze the expected time interval from the early phases on PDP (note that Figs. 2.5 and 2.8 do not include time), considering both the SPD and the IPD, the time interval needed to accomplish the conceptual phase in the latter is greater than in the former (Fig. 2.9). This is the consequence of the early exchange of information among the different areas which aim to reduce the total design and development lead time.

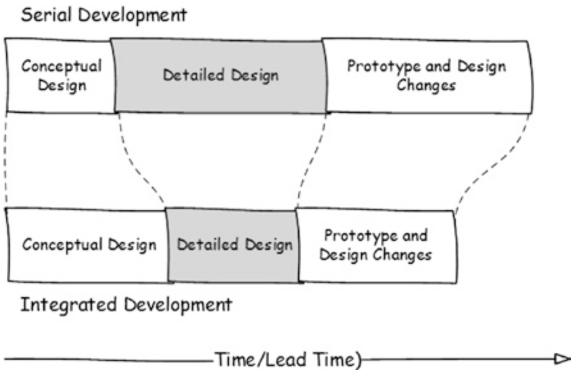
While major design changes were anticipated and solved at the conceptual phase where the changes are easier and cheaper to implement, some design changes might still occur at the IPD remaining phases. However, these changes are significantly less important than those discussed and solved at the conceptual phase.

These duration's expectations, as shown in Fig. 2.9, pose a managerial dilemma of IPD. How could a company be certain about the expected decrease of the product development lead time?

Take, for instance, the aeronautical sector. Even though there is public data about expected lead time reductions from 48 to 36 months during aircraft development, the manager will have to wait 3 years before becoming certain of the IPD investment return. In the meantime, the manager will receive all the pressure to present tangible results, while keeping a "*Festina lente*" (make haste slowly) attitude. Overcoming this managerial dilemma is one of the challenges for a company which is used to the serial approach to change into integrated product development.

A way to surmount the managerial dilemma is to select a pilot development of a product that is simpler and quicker to implement than the company's main product. For instance, an aircraft manufacturer could choose a fuel tank to initiate the

Fig. 2.9 The integrated product development managerial dilemma



IPD approach. After gaining confidence during this pilot project, the experience and knowledge could be extrapolated to other systems and then to the company's main product.

There are, essentially, two main resources required to implement the IPD approach, namely, multifunctional design teams and IPD tools.

2.5 Integrated Product Development Teams

Suppose you look at a bus stop and see a gathering of people. I what you see a group or a team? Certainly, it is a group of people because one person might go to the town center, another to suburb, and so forth. Then, some people get on the bus heading to the town center. Are the people inside the bus a group or a team? Certainly, it is a group as one person might stop by the library, another to train station, and so forth. Finally, you see some people from that bus coming out of it to try to fix an engine breakdown. Are the people trying to fix the engine a group or a team? Certainly, it is a team. From this simple example one might figure out that the four characteristics of a team are: mission, commitment, complementary capacity, and ephemerality as shown in Fig. 2.10.

The mission of the IPD team is to assure that the requirements of all product development phases are evenly represented in the IPD's conceptual design phase. All people from the IPD design team should be committed to obtaining the best possible balanced results for the product, even if that means giving away some of his/her technical area expectations [2]. Complementary capacity is achieved by having representatives from all PD technical areas on the IPD team, such as: marketing, design, manufacturing, assembly, maintenance, etc. An IPD design team is ephemeral because after finishing a given product development, that particular IPD team ceases to exist.

Ideally, all technical areas from the product lifecycle phases are represented in a typical design team meeting and a number of engineering tradeoffs are raised, discussed, and solved. For instance, the choice of a certain electric spindle for

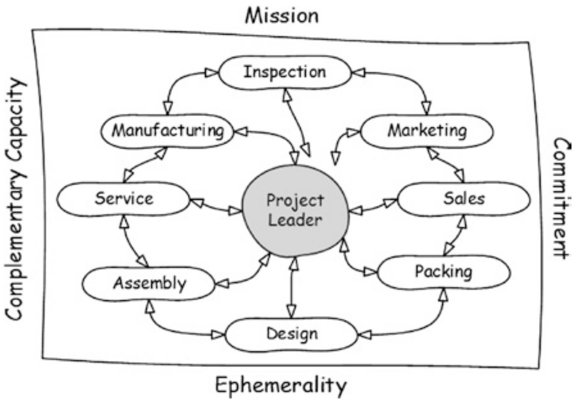


Fig. 2.10 Multifunctional design teams

a robot end effector might suit the power requirements for the drilling operations. On the other hand, that very spindle could jeopardize the weight payload of the robot. In another example, a person from manufacturing would argue that the product geometry would be better that way in order to avoid reorientation of the part. A person from marketing would argue that the geometry just proposed by manufacturing would not sell a piece. The adequate spindle should suit both requirements. At the end of the meeting, nobody leaves either "100 % happy" or "100 % unhappy."

It is the role of the project leader to ensure the team's focus on the mission and achieve a balanced result. The specialists, while they have deep vertical knowledge in their subjects, must have the maturity to explore the horizontal knowledge which is how each subject can interface in order to leverage the others and benefit the product as a whole (Fig. 2.11).

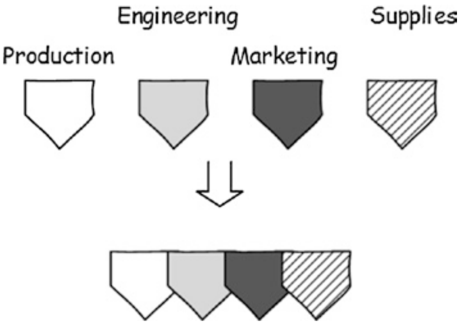


Fig. 2.11 Vertical and horizontal knowledge in IPD design teams

2.6 Integrated Product Development Tools

This section presents a quick overview of some IPD tools. We show the benefits brought from integration to the product development process without going into much detail about the step-by-step use of the tools.

2.6.1 Design by Features (DbF)—as a Potential IPD Design Tool

Design by Features (DbF) is a CAD resource; although it is not an IPD tool, it gives a good example of possible tool adaptation to support the cooperation among the design and the manufacturing teams.

DbF was developed in the early 1990s to replace the cumbersome way of drawing manufacturing features such as holes, pockets, edge fillet, and so forth in a CAD design. Prior to DbF, the CAD designer had to make a Constructive Solid Geometry (CSG) approach, drawing two solids, a block and a cylinder, aligning them, and making a Boolean subtraction to draw a hole (Fig. 2.12a). Alternatively, he/she could use the Boundary Representation (BRep) approach by drawing the whole surfaces and assembling them afterwards (Fig. 2.12b).

For drawing the same hole using the DbF approach, a CAD designer chooses the feature <hole> from a drawing pallet, defines the hole type, for instance, <blind> as well as the hole dimensions: <diameter> and <depth> as shown in Fig. 2.13 and indicates the place the hole should be on the workpiece.

This is a pure CAD task assisted by DbF. Suppose that the engineer or designer chooses the hole's diameter as 10.37 mm. The CAD will draw the holes all the same. When the CAD file is sent to manufacturing shop floor, the drill operator will not find a commercially available 10.37 mm drill for drilling the hole as specified by engineering. The drill operator does not have the authority to decide upon a different hole's diameter based upon the commercially available drills. Therefore, a communication protocol has to be established between design and

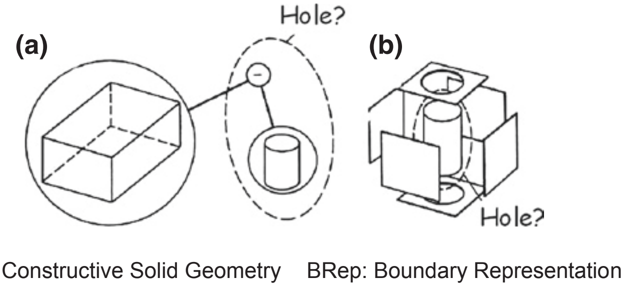
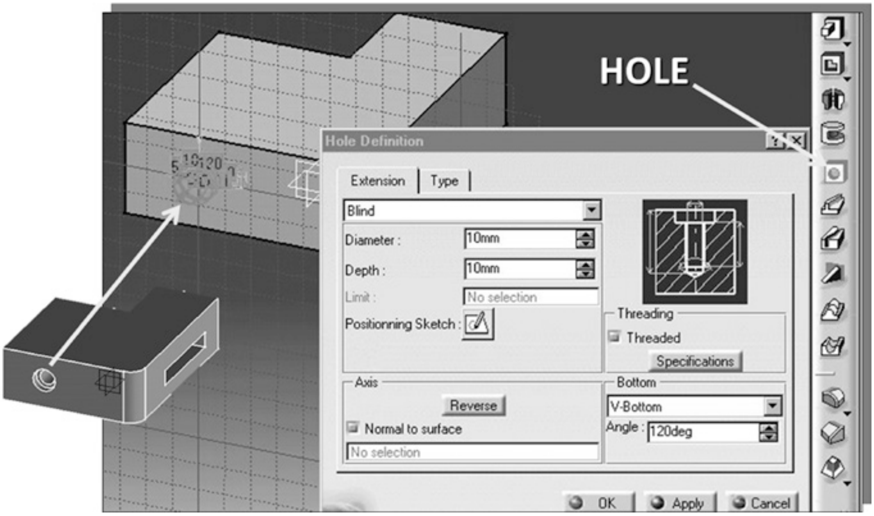


Fig. 2.12 CAD approaches for drawing manufacturing features. **a** *CSG* Constructive solid geometry, **b** *BRep* boundary representation



Blind

Diameter: 10mm

Depth: 10mm

Limit: No selection

☐ Threaded

☐ Normal to surface

No selection

V-Bottom

Angle: 120deg

Fig. 2.13 Drawing a hole using the DbF approach

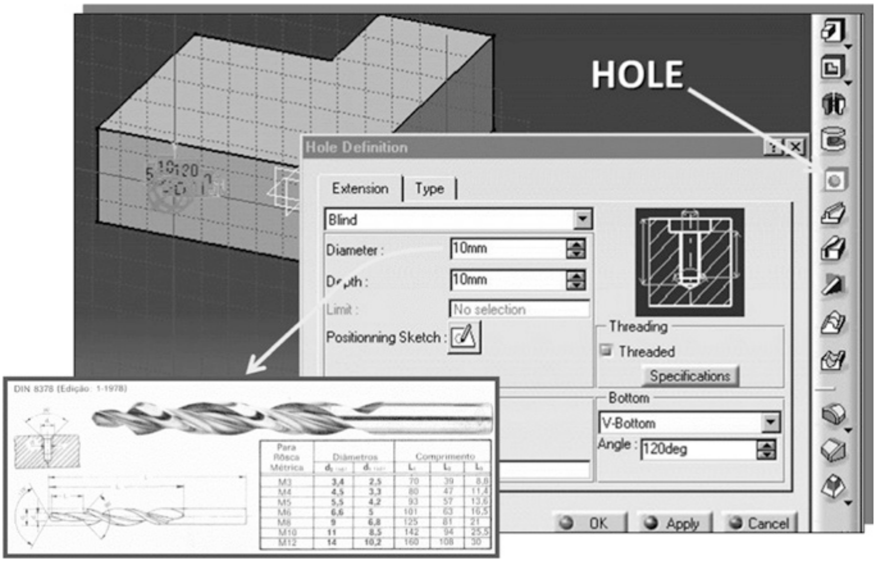
manufacturing areas in order to decide the final diameter of the hole. This value could even be 10.37 mm, but it will require a special drill that costs more than the commercial counterpart. Nevertheless, this is an example of oscillation in the product development process characterized by a loop that consumes valuable time without adding value to the product.

Suppose the engineer or designer is presented a set of commercially available drills, as soon as he or she selects the <diameter> scroll bar in the CAD screen as shown in Fig. 2.14.

The IPD-DbF design tool has the <diameter> scroll bar locked to a drill table so that the designer is required to choose one of the commercial available drills. This is a true design-manufacturing integration accomplished in a very efficient and clever way as the designer does not leave his/her work environment to search for information, and the drill operator will do his/her job as soon as he/she is required to. At the very end of that table, after all commercial available drill choices, a blank field can be shown to deal with special cases such as the 10.37 mm drill.

2.6.2 Knowledge Based Engineering (KBE)—a Truly IPD Design Tool

KBE is a computer-based design environment where the design intent can be captured, executed, and disseminated through a company. Suppose an engineer is given the task of dimensioning spars and ribs of an aircraft wing as shown in



Hole Definition

Extension | Type

Blind

Diameter : 10mm

Depth : 10mm

Limit : No selection

Positionning Sketch :

Threading

☐ Threaded

Specifications

Bottom

V-Bottom

Angle : 120deg

OK Apply Cancel

DIN 8378 (Edição: 1-1978)

Para Rôsca Métrica	Diâmetros		Comprimento		
	d_2	d_1	L_1	L_2	L_3
M3	3,4	2,5	70	39	8,8
M4	4,5	3,3	80	47	11,4
M5	5,5	4,2	93	57	13,6
M6	6,6	5	101	63	16,5
M8	9	6,8	125	81	21
M10	11	8,5	142	94	25,5
M12	14	10,2	160	108	30

Fig. 2.14 DbF adapted as an IPD design tool

Fig. 2.15 Spars and ribs dimensioning of an aircraft wing

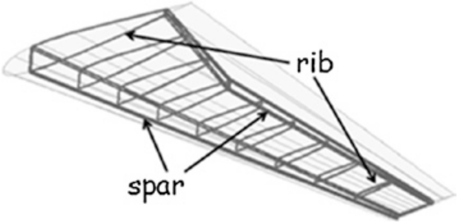
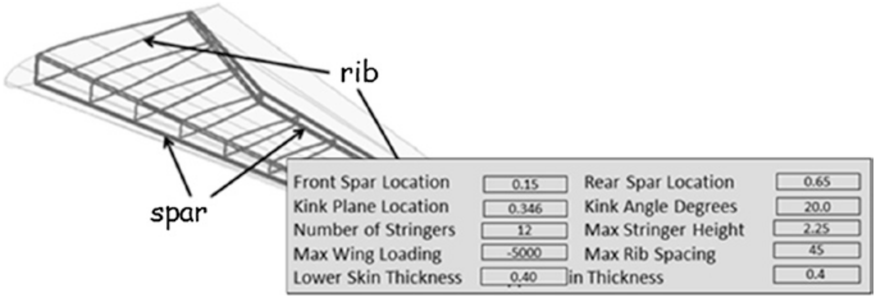


Fig. 2.15. Both the leading edge spar and the trailing edge spar are…spars! Nevertheless the engineer has to repeat the dimensioning procedure for both spars taking into account the differences of geometry, load conditions, and assembly docking features. The same situation happens for the 10 ribs shown in the picture.

To overcome the burden of repeating the dimensioning procedure as exemplified, KBE turns the definition of a specific product design process into creating rules, activities, and decisions that a skilled engineer would follow to accomplish the product dimensioning and design. Therefore, the written design procedure would be created by a person (the same engineer who wrote it or somebody else) and all the spars and ribs would be dimensioned, designed, and drawn. The creation of the written procedure looks like a CAD parametric window (Fig. 2.16), where the user inputs some information about the part he/she wants to design, such as part location, space among the parts, maximum load upon a part etc.



Front Spar Location	0.15	Rear Spar Location	0.65
Kink Plane Location	0.346	Kink Angle Degrees	20.0
Number of Stringers	12	Max Stringer Height	2.25
Max Wing Loading	-5000	Max Rib Spacing	45
Lower Skin Thickness	0.40	in Thickness	0.4

Fig. 2.16 Spars and ribs dimensioned automatically by the KBE engine

The KBE design environment is like a blank sheet of paper where a specialist applies his expertise of how to design and dimension a given part. The key part of KBE is named *generative model*, where the design intent, dimensioning procedures, rules, tables, and other contents can be linked to finite element analysis, cost analysis, manufacturing and assembly restrictions, guidelines, and evaluations. Due to this additional content, KBE has the potential to establish integration among the technical areas of the product life cycle. The dialogue between design and manufacturing, for instance, might be accomplished through a written procedure of the generative model of KBE. This is exemplified by a practical industrial case depicted in Fig. 2.17.

In the serial tube design and manufacturing approach (Fig. 2.17a), the designer draws a 3D tube to meet functional requirements such as to connect two ends of the air conditioning system. To accomplish that, a series of bends and curves need to be drawn and modeled. After finishing the design and modeling of the tube, a manufacturing engineer checks whether the bending machine is capable of bending the tube with the angles specified by the designer. If all the angles are feasible to be manufactured, the part number is approved; otherwise, the 3D drawing of the tube returns to the designer who, by his/her turn, corrects the angles. Once the part number is approved, a set-up operator inputs the necessary data to run the bending machine. Then the bend machine operator finishes the process and the tube is ready to be installed. It's clear that this design process has several opportunities for improvement, mostly related to the elimination of the design oscillation phenomenon already described herein.

An integrated tube development approach based upon the KBE engine is pictured in Fig. 2.17b.

In this approach, a KBE engine has been developed and placed at the workbench of the designer. As the designer starts drawing a tube section, the KBE engine checks (online) the angles drawn by the designer against the manufacturing parameters that are based upon the capability of the bend machines and signals to him/her the necessary corrections to be implemented. The 3D tube leaves the CAD workstation only when it is ready to be manufactured. The tube data are filled and exported to the bending machine that finishes up the tube. The design-manufacture

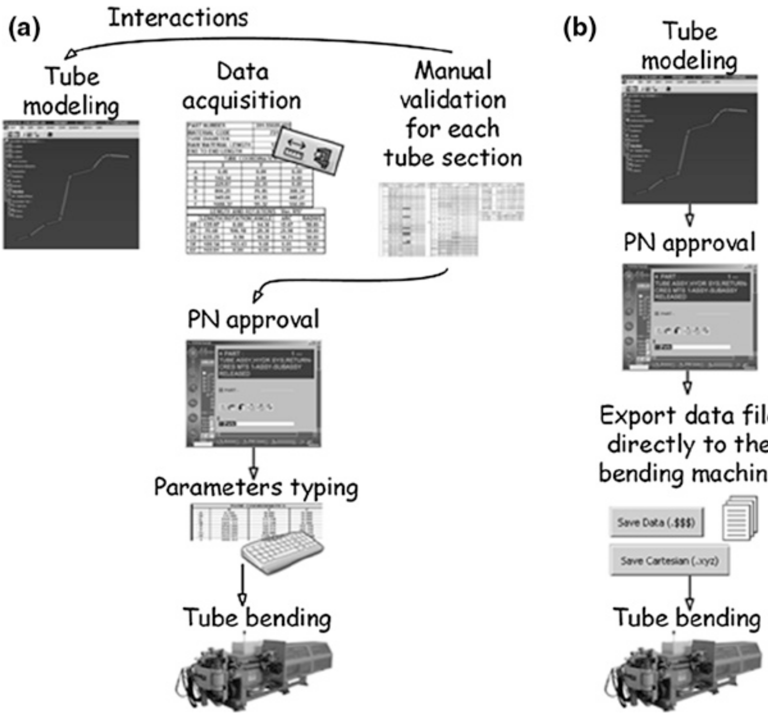


Fig. 2.17 Serial (**a**) versus integrated (**b**) tube design and manufacturing based upon the KBE engine

loop from the previous process has been replaced by a virtual and efficient dialogue between the designer and the manufacture engineer based upon the KBE engine.

It's worth noticing the differences and similarities between the DbF and KBE: a CAD environment is indispensable for both; the interaction between design and manufacturing is executed by the designer in DbF while it is rule based in KBE, without the designer interference. The majority of IPD tools, though, do not require a computer environment in order to promote the required integration.

2.6.3 Design for Excellence (DFX)

A great number of design tools are available to promote the integration of the technical areas of the product development, such as Design for Manufacturing (DFM), Design for Assembly (DFA), Design for Recycling (DFR), Design for Service (DFS), Design for Packing (DFP), Design for E-Business (DFEB), Design for Automation (DFAut), and so forth.

All these DFX (Design for X or eXcellence, where X can be thought as a variable that can undertake the "values" M, A, R, S, P) have in common the aim to integrate the requirements of the technical area X into the conceptual design phase of the product. [3] The DFX design tools are indeed tools to be used by the IPD team members to advocate the best design option for the product regarding their technical areas. Among all possible product design options, the manufacturing area, through DFM, will point out those that best fit the manufacture requirements. Among all possible product design options, the maintenance area, through DFS, will point out those that best fit the maintenance requirements. It is quite possible that the DFM product option conflicts with that of DFS, thus raising an engineering tradeoff whose solution might partially fulfill both areas.

Some DFX IPD design tools are already well known and consolidated such as Design for Manufacturing (DFM), Design for Assembly (DFA) and Design for Service/Maintenance (DFS). Others are proposals yet to be tested such as Design for E-Business, Design for Nationalization, Design for Patent. All of them are related to some phase of the product life cycle and have one characteristic in common: an attempt to integrate the requirements of their product life cycle phase into the conceptual design phase of the product development process. It's worth stressing the words "attempt to integrate" because all the representatives of product life cycle phases will try to do the same—to advocate their cause. If just one phase or technical area prevails, the final configuration of the product would resemble of those shown in Fig. 2.18.

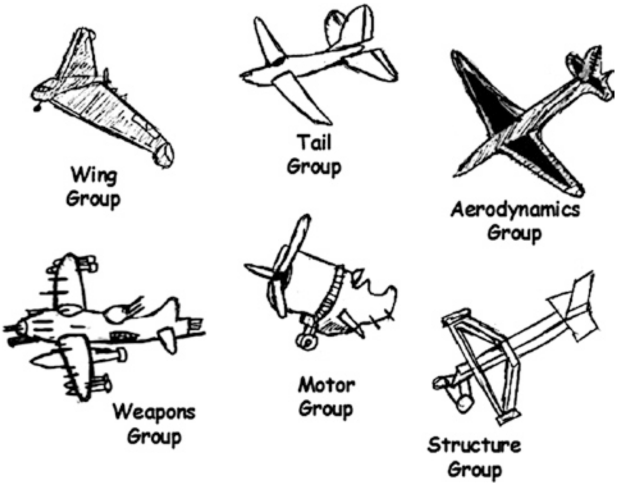


Fig. 2.18 Nonintegrated product development

In order to avoid this situation, the technical coordinator of the IPD team has to assure that all the requirements are taken into account, discussed, and incorporated into the product design in a balanced manner. It is correct to say then that in an IPD design team meeting, nobody leaves it 100 % happy and nobody does it 100 % unhappy.

Some DFX tools are implemented through guidelines which are derived from the merge of the design experience with the experience a person or a team has had in an “X” area. Some DFX guidelines are implemented through a systematic approach or method. One of them is the Design for Assembly (DFA). The guideline “Design for minimum part count” has been converted into a method by Boothroyd and Dewhurst in 1981 [4] and has evolved since then.

The Boothroyd and Dewhurst method for finding the minimum number of parts a product must have is based upon three questions the designer or team has to answer for each part or component which belongs to the product structure.

1. Does the component move relative to all other components already assembled?
2. Must the material of the component be different from those of the other components already assembled?
3. Must the component be separate from the other components already assembled to give access or disassembly them?

It should be noted that the causes for answering “Yes” to questions (1) and (2) must be related to the product’s functionalities. A movement of a screw when it is been screwed receives a “No” answer for question (1). However, a power screw from a press has “Yes” as answer for the same question. Question (2) takes “Yes” for an answer whenever the component is used as electrical, thermal, or acoustic isolation, for instance.

Based upon the three questions, one must conclude that the minimum number of components or parts for *any* product is:

Minimum no. parts = No. of parts which has at least one “Yes” irrespective of the question + 1 (the prime or base part).

Figure 2.19 shows one product assembly example, before applying the method. The total number of components of this product is 20: four main components and 16 components for assembling the main components into the final shape. Also, the axle shown moves relative to the base due to the gear rotation.

If you go through the three previous questions, you conclude that the minimum number of components is two as shown in Fig. 2.19a.

The main benefit of the DFA analysis is not to determine the minimum number of components but rather, to search for new design proposals for the product that take into account the minimum number of parts. Suppose the design team has proposed the product design shown in Fig. 2.20a. Probably, the designer does not have the necessary information about the stamping process needed to manufacture the proposed design. Then it is sensible to think that he/she will ask the stamping specialist if the product “as designed” could be transformed into an “as built” product. By doing so, the DFA analysis fulfills its main objective, i.e. to promote the integration between design and manufacturing/assembly.

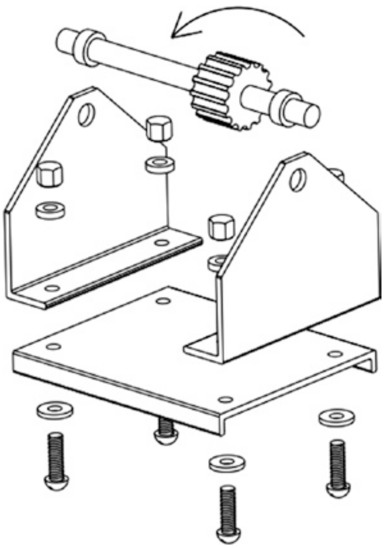


Fig. 2.19 An example of a product assembly

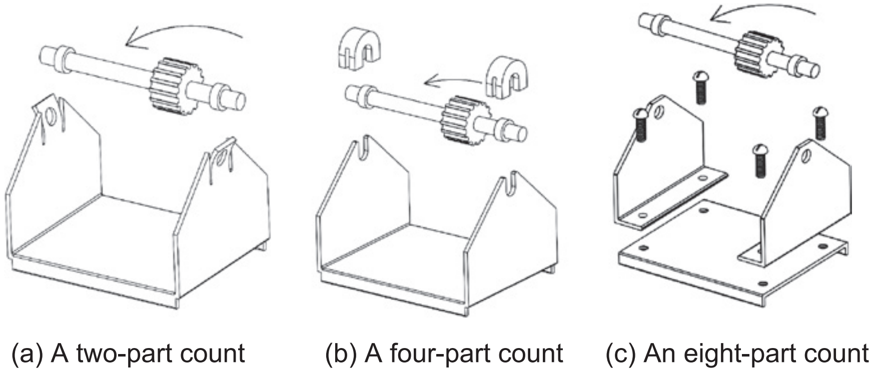


Fig. 2.20 Product redesign after the DFA analysis. **a** A two-part count **b** A four-part count **c** An eight-part count

The outcome of this design-manufacturing meeting could have been: our company hasn't have the necessary press to stamp the shape of the base or the volume of sales of the product does not justify the investment in a more complex stamping die.

The dialogue goes on. What about the design shown in Fig. 2.20b? It does not meet the minimum of part criterion but this is not the main issue. However, the stamping specialist still finds the stamping die rather complex and asks for an alternative product design. Finally, the design option depicted in Fig. 2.20c is the one that meets–partially–the design and manufacturing requirements.

2.6.4 Integrative Design Variables (IDV)

Costs, weight, center of gravity, and net electric power are examples of integrative design variables. The characteristics of these variables are the following:

1. There is a **target value** associated with them within a specific product development. Examples: the cost of an aircraft cannot be greater than $14.5 M; the maximum weight of a robot end effector is 80 kg; the net power of a satellite is 2300 W.
2. These variables are **affected by almost all design decisions**. Examples: the choice of a single component impacts cost, weight, center of gravity, and perhaps net electric power if the component requires it for operation.
3. It is **easy to grasp the concept** around integrative design variables. Design people do not need to be lectured about them as their understanding is quite straightforward. Examples of design variables that do not meet this characteristic of IDV are: aerodynamic drag, wear, and stiffness.

The IDP design tools that address the integrative design variables are named DTX—Design *to* X rather than DFX—Design *for* X. Design to Cost (DTC), Design to Weight (DTW) [5], Design to Net Power (DTNP), Design to Center of Gravity (DTCG) are examples of the former.

Suppose that the IDV are displayed together within a specific product development and are regularly updated by the methods related to each of them. The resulting scenario is a technical managerial cockpit shown in an illustrative form in Fig. 2.21.

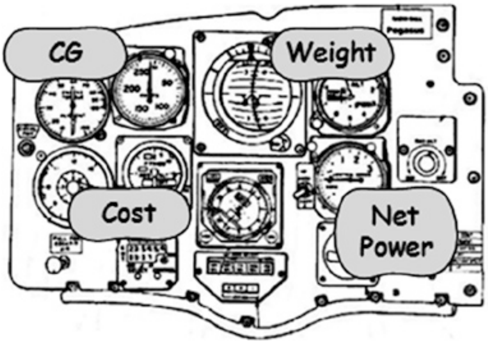


Fig. 2.21 Technical management cockpit

2.6.4.1 Design to Cost (DTC)—an Introduction to Integrative Variables

The basic equation that drives DTC is the following:

$$TC = TSP - TP \tag{2.1}$$

where:

- TC = Target Cost of a product
- TSP = Target Sale Price of a product
- TP = Target Profit of a company

The TSP variable is the starting point for DTC and it is usually obtained by the market intelligence department of a firm. TP is a firm internal variable that is usually set by its stakeholders. Then a product to be sold cannot cost more than TC.

DTC is also based on a process consisting of the following steps:

Step 1: **Establish the product requirements**. The design team may use some well-known design tools or methods to accomplish this step such as the Objective Tree method. In this method, a preliminary generic need is unfolded in several levels up to a stage where the need is converted into more meaningful and precise statements or requirements.

Step 2: **Define the functional structure of the product**. A design method that might help the design team work in this step is the Functional Analysis method. Similar to the previous method, Function Analysis is based upon a deployment activity, an overall, "black box"-like function is deployed in sub functions. The black box is transformed into a "transparent box."

Step 3: **Elaborate design alternatives for the product**. The Morphological Chart can be used to carry out this step.

Step 4: **Estimate the cost of the functions**. DTC prescribes the comparison the cost of the functions rather than the cost of the whole product. In doing so, the design team has more strict control over the design decision to meet the target cost of the product. The estimate cost of the functions is obtained from the matrix shown in Fig. 2.22.

The elements required for filling in the rows and columns of the matrix are the results from steps 2 and 3, respectively. Additionally, the design team has to search for the cost of the components required to fulfill the functions as indicated in the last line of the matrix. The remaining variables of the matrix are defined as follows.

$\mathbf{V_{ij}}$ = binary variable that indicates whether there exists a relationship between the component $\mathbf{A_j}$ and the function $\mathbf{F_i}$ ($V_{ij} = 1$) or not ($V_{ij} = 0$)

$\mathbf{a_{ij}}$ = variable that indicates—percentage—how much the component $\mathbf{A_j}$ influences the performance of the function $\mathbf{F_i}$

$\mathbf{Z_{ij}}$ = partial cost of the function $\mathbf{F_i}$ with regards to the component $\mathbf{A_j}$ obtained as:

$$Z_{ij} = V_{ij} \times a_{ij} \times C_{Aj} \tag{2.2}$$

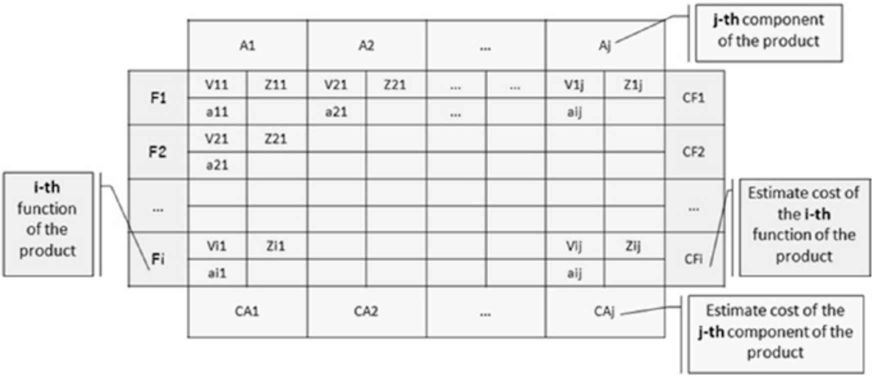


Fig. 2.22 Estimate function cost matrix

Finally, the estimate cost of the function F_i is calculated as presented in Eq. 2.3:

$$C_{Fi} = \sum_{j=1}^{k} Z_{ij} \tag{2.3}$$

Equation 2.3 must be repeated for all **k** functions present in the functional structure of the product.

Step 5: **Compare the Estimate Cost of a Function (ECF) with the Target Cost of a Function (TCF)**. In order to accomplish this step, the design team has to figure out the target cost of the functions based on the target cost of the product. This can be achieved through the Value Analysis Method, where a survey must be carried out with the customers of a given product to determine how valuable the functions are to them. The results could be in a qualitative rank such as Low, Medium, or High.

Two possible results might come out from the comparison between the two categories of function costs:

$$\text{ECF} < TCF \tag{2.4}$$

or

$$\text{ECF} > TCF \tag{2.5}$$

The result presented by Eq. 2.4 is favorable for the design team and some actions could be derived from that so as to improve the components quality of this function or add some complementary sub functions to the main function. However, the technical coordinator of the IPD must be aware of what function could be causing the result from Eq. 2.5. Some actions that could be driven from that are to withdraw some sub functions or replace the components with lower cost counterparts.

Step 6: **Optimize the conceptual design of the product**. Instead of reasoning in an isolated manner, looking at the functions individually, the IPD design team could try to balance out the results from Eqs. 2.4 and 2.5—the surplus of one function could be used to rescue the deficit of another function. The scenario just described is an example of an IPD meeting agenda where one specific technical area, for instance, structures, meets another area, such as interiors, to sort out the best possible balanced solution that meets the design needs of as well the function target cost of both areas.

If you are considering an IDV different from cost, the logic from the previous steps remain the same, only substituting the measured variables.

2.7 A Practical View

A very short list of IPD tools have been shown and discussed. Those IPD tools not presented herein are not less important than those tools presented herein. Integrated design methods such as QFD—Quality Function Deployment [6, 7], Design for Environment [8]; Design for Service [9] have to be applied to the product design so that all technical areas of the product life cycle are represented and heard at the conceptual design phase of the product development process.

Keep in mind that the final result of any DFX or DTX technique is the product. Take, for instance, an academic assessment as the product. Suppose you need to prepare an assessment about the subject Lean Product Development (LPD). The ordinary way to prepare it is to quickly define open questions such as “Discuss about the impact of LPD over the ISO 9000 certified companies.” It’s easy to think of nine more questions similar to that. However, the whole assessment process includes the correction and marking of the assessments. That can take a lot of time that is directly proportional to the number of students.

Design for Correction is an application of the DFX techniques to academic or technical assessments [10]. The assessment (that’s the product) takes into account the requirements of the correction process as well. Naturally it takes longer to elaborate upon the questions compared to the traditional way, but the whole assessment process is shorter because the correction can be done as shown below.

The DFC questions are prepared in a way that requires the students to establish the relationship among a number of concepts, approaches, and techniques discussed during the course. The students are free to check their class notes, slides, books, and papers. A sample question is shown below.

Mark the correct alternative(s) with regards to Lean Development Product (LPD):

(a) It is more important to understand how the Lean philosophy is applied to the Product Development Process (PDP) than to know the lean techniques and tools.
(b) Because LPD radically differs from Integrated Product Development (IPD), it makes LPD a very difficult matter to be understood by western companies.
(c) As Knowledge Management (KM) is a weak characteristic of LPD, the two subjects are complementary and create a sustainable and competitive advantage for the companies.
(d) The continuous improvement associated with LPD has little impact over the PDP performance indicators once the majority of the companies are already ISO 9000 certified.
(e) Based upon the triad Knowledge, Skills and Attitudes (KSA), the teaching of LPD in western companies has to be focused on Knowledge.

The open question "Discuss about the impact of LPD over the ISO 9000 certified companies," is replaced by five alternatives with more strict content. Nevertheless, in all the alternatives the student has to review several concepts and the relationship among them. In the sample question, the concepts of LPD philosophy, LPD tools, IPD, KM, KSA are intentionally mixed.

The test lasts 60 min and the students keep their test sheets for correction. The lecturer starts the oral correction by stating the correct answer for each question (answer "A" on the sample question). A student might argue that another answer is also correct; having to explain what sustains his/her choice. Other students might join the discussion and turn the correction process into a "Greek Agora Square." Naturally, the lecturer has to keep the discussion under control, avoiding the corporatism syndrome. Eventually, the argumentation of the student might be taken into account and the lecturer would consider the student's choice correct. It is not the case, though, for the presented sample question.

References

1. Andreasen MM, Hein L (1987) Integrated product development. IFS-Springer Verlag, Berlin
2. Paris C, Salas E, Cannon-Bowers J (2000) Teamwork in multi-person systems: a review and analysis. Ergonomics 43(8):1052–1107
3. Huang GC (1996) Design for X: concurrent engineering imperatives. Chapman&Hall, London
4. Boothroyd G, Dewhurst P (2010) Product design for manufacture and assembly, 3rd edn. CRC Press, New York
5. Furtado LF, Villani E, Trabasso LG, Silva CEO (2013) DTW: a method for designing robot end-effectors. J Braz Soc Mech Sci Eng. doi:10.1007/s40430-013-0109-8
6. Akao Y (1995) Quality function deployment: integrating customer requirements into product design. Productivity Press Inc., New York

7. Araújo MF, Trabasso LG (2013) Applying QFD to business development environment. J Braz Soc Mech Sci Eng 35:1–14. doi:10.1007/s40430-013-0010-5
8. Guerato AM, Araújo CS, Trabasso LG (2010) A method to obtain context based DFE criteria lists applicable to the development of friendly recycling composite aeronautic structures. Product: Management & Development (IGDP), v.8, pp 71–79
9. Vezzoli C, Kohtala C, Srinivasan A (2014) Product-service system design for sustainability. Greenleaf Publishing
10. Pessôa MVP, Trabasso LG (2014) A Lean way to teach Lean Product Development to Graduates. In: IIE Engineering Lean and Six Sigma Conference and the Lean Educator Conference: Orlando, USA