**INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09**
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

# FUNCTIONAL BASIS AND B-CUBE: ALTERNATIVE OR COMPLEMENTARY MODELS?

**Vicente Chulvi[1] and Rosario Vidal[1]**
(1) Engineering Design Group (GID), Dpt. of Enginyeria Mecànica I Construcció, University Jaume I, Castellón, Spain

## ABSTRACT
Human beings have always tried to classify knowledge in order to manage it. A common way to structure knowledge in a hierarchical manner is by means of taxonomies. Within the context of engineering and functional design several function taxonomies have been developed. The most significant reconciliation of function taxonomies nowadays are those provided by the National Institute of Standards and Technology, called the functional basis. A new model for managing knowledge, the B-Cube model, has been developed based on the main achievements of the National Institute of Standards and Technology, namely, reduced ambiguity and increased uniformity.
The present work explains, in general lines, the B-Cube model developed by the authors, and it shows a comparison between this model and the functional basis. The aim of the paper is to defend the usefulness of B-Cube in functional design within the Function-Behaviour-Structure framework, and to demonstrate that it can fit the Behaviour level and work together with the functional basis.

*Keywords: FBS framework; B-Cube; functional design; functional basis*

## 1 INTRODUCTION
The amount of knowledge in the universe is huge, almost infinite, but human beings have always tried to manage it. Every branch of science has developed its own ways to achieve this objective, and many vocabularies have been elaborated in order to classify a specific area of knowledge. A specific kind of classification commonly used is known as a taxonomy. A taxonomy consists of a group of concepts and relationships that are organized hierarchically and whose concepts can be arranged as classes with sub-classes [1]. Taxonomies were introduced into the industrial world by Gershenson and Stauffer [2], but Szykman et al. [3] were the first to differentiate functions with their extensive review of function terminologies within the engineering context between 1976 and 1998. Since then, several function taxonomies have been developed [4,5], although the most significant reconciliation of function taxonomies nowadays are those provided by the National Institute of Standards and Technology (NIST) [6].
The NIST functional basis is a reconciliation and integration of other independent research efforts [7-9] in order to develop a formal representation of functions, i.e. a taxonomy of standardized terminologies, focused on mechanical design. The NIST's work stems from three specific needs: the representation of functions in Computer Aided Design (CAD), a fixed scheme for modelling functions, and a universal set of functions performed by mechanical systems. The greatest achievements reached by NIST are reduced ambiguity and increased uniformity. To reduce ambiguity they defend that the more terms are used to refer to the same concept (synonyms), the greater the number of different ways to model a given concept there will be. Increased uniformity attempts to facilitate the exchange of function information among different applications.
These two concepts are of great importance when working with functions in design. But functional design depends on more than just the function level. The cornerstones of functional design are the concepts of Function, Behaviour and Structure, which, due to their functionality, were defined and proposed as a framework for modeling and representing by Gero [10]. The Function-Behaviour-Structure (FBS) framework is widely used among designers as a methodology for design process analysis, as it represents the evolution of the design state from the study of protocols [11]. Within this framework, Function represents the duties that the design fulfils, Structure represents the physical elements of the solution, and Behaviour is the link between F and S. In solution synthesis, Behaviour

derives from a particular Function, and the solution is achieved through this Behaviour. Furthermore, when a solution has been defined, Behaviour is inferred from that solution in order to evaluate whether the solution reaches the required degree of functionality. More recently, Gero has extended the study on FBS representation [12,13], as have a number of other authors [14-22].

As different levels appear on functional design, besides Function, taxonomies for these other levels have been the object of study of other authors. So, several Behaviour ontologies have been developed [23-26], but maybe the most interesting study for the development of Behaviour ontology is the DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) [27]. However, strictly speaking, the DOLCE is not a taxonomy. It is considered to be an upper-level ontology. An ontology can be described as an explicit specification of a shared conceptualization, where concepts and relations are organized hierarchically and concepts are classified as classes and instances [28,29]. Thus, upper-level ontologies describe very general concepts (e.g. substances, tangibles, intangibles) and provide general notions under which all root terms in existing ontologies should be linked.

The aim of this paper is to compare the NIST functional basis and the B-Cube model. B-Cube, or Behaviour's Cube, [30,31] attempts to cover the knowledge organization of behaviours within the FBS framework, and it is broadly outlined in Chapter 2. Chapter 3 discusses the differences and similarities between the functional basis and B-Cube. Lastly, in Chapter 4, the conclusions from this comparison are presented.

## 2 B-CUBE MODEL

B-Cube (represented in figure 1) is a new approach to functional design through the FBS framework. This approach proposes a three-dimensional scheme that uses definitions as Behaviours concepts. The key to this approach is that a Behaviour is not defined with a word or a taxon, which could cause ambiguity and misinterpretation, but rather it is defined as a three-dimensional vector (X, Y, Z) that is set by its characteristics and qualities.

The starting point from which to define these parameters is the DOLCE's upper-level onthology. The B-Cube model therefore requires some definitions from the DOLCE, with a few adaptation of their meaning. *Endurant*, for example, is defined as the entity or element (Structure) to which the B-Cube entry refers. It is supposed that there are an infinite number of endurants in the universe, and they are differentiated as being *physical* (PEDs) and *non-physical endurants* (NPEDs). *Perdurant* (P) is a characteristic that defines a Behaviour, and it refers to the kind of behaviour that affects the above-mentioned entity. Ps are situated on the Y axis of the B-Cube model. Lastly, *qualities* are defined as characteristics linked to other entities, which are going to be used to define the Behaviour. There are three different sorts of qualities: *temporal* (TQs), *physical* (PQs), and *abstract qualities* (AQs). TQs are directly related to Ps, so they will be used to define a Behaviour, and are located on the Z axis. The B-Cube model is completed with the X axis, where the PQs will be, if the entry to the model was a PED, and AQs, if the entry was an NPED.

Despite the fact that numerical values are used when working with the B-Cube model, all of these values have a term to define them. The terms have been taken mainly from the DOLCE's terminology [27], and increased by Garbacz's work on it [32]. As these terms were not enough to complete the needs of B-Cube, they were fulfilled with Rasmussen's taxonomy terms [23] and the NIST classification of flows [6,33]. As a result, they are defined as follows:

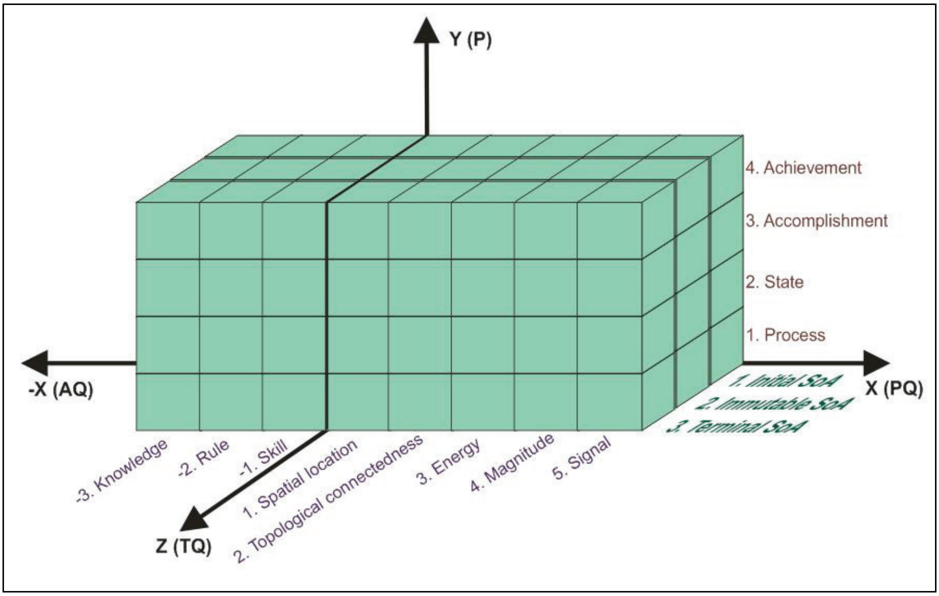


*Figure 1. B-Cube model*

<u>X-Axis:</u>
Positive values. These correspond to the definition of PQ in the DOLCE's ontology. Only one definition is provided within this group (*spatial location)*, but Garbacz increases the number of PQs to three (with the addition of *topological connectedness* and *energy*). According to the treatment of flow and functions in a Black Box model by Nagel [33], it can be deduced that two more terms are needed in order to define all the flows that can interact with a function (i.e. *magnitude* and *signal*). So, these values are defined as follows:

- 1: Spatial location. This is related to the position of a PED in space. Moving an object belongs to this category.
- 2: Topological connectedness. This concerns the sort of connection at the topological level on which the PED is located. Breaking or joining an object corresponds to this group.
- 3: Energy. This refers to the energy state of the PED. Freezing water or charging a battery are examples of Behaviours classified within the *energy* group.
- 4: Magnitude. This is related to the physical magnitude of the PED that is affected by the P. Increasing the weight or changing the color of an object are examples that correspond to this group.
- 5: Signal. This is related to actions involving PEDs when they act as signals. Examples could be increasing a wave or a cell-phone that is sending a signal.

Negative values. These correspond to the definition of AQ, according to the DOLCE. As there are no values defined for AQs, the terms used in Rasmussen's taxonomy for human behaviours are used here:

- -1: Skill. The Behaviour does not need any conscious control by the subject. Driving a car or playing the piano are examples of *skill* (although this is not so during the learning process).
- -2: Rule: The Behaviour needs conscious control by the subject, but this is limited by some kind of process or "written" rule. Cooking following a recipe or tuning an instrument are examples of *rule*.
- -3: Knowledge: The Behaviour needs conscious control by the subject, and this is not limited by any kind of process or "written" rule. Composing a symphony or managing an enterprise are examples of *knowledge*.

<u>Y-Axis</u>

The values of P, as defined in the DOLCE, are placed on this axis. This set of Ps seems to cover all the needs in this respect in B-Cube. In order to gain a better understanding of the meaning of each of these terms, some definitions are needed. The first, cumulative, is when the mereological sum of two cases of the same type remains that same type. That is, to run is cumulative since “to run” and “to run” results in “to run”. Nevertheless, to give a conference is non-cumulative since “to give a conference” and “to give a conference” results in “to give two conferences”. Homeomeric is when all the temporal parts are described by the very same expression used for the whole occurrence. So, “to sit” is homeomeric because every instant of the action can be defined as “to sit”, while “to run” is non-homeomeric because its instantaneous actions can be expressed as “left foot”, “right foot”, “left foot”, and so forth. Lastly, atomic is when the case is immeasurably short in time. Thus, to break a pane of glass is atomic ($t \rightarrow 0$), while to give a conference is non-atomic ($t \approx 20$ min). So, P values are defined as:

- 1: Process. The Behaviour is cumulative and non-homeomeric, like the example “to run”.
- 2: State. Cumulative and homeomeric, like “to sit”.
- 3: Accomplishment. Non-cumulative and non-atomic, like “to give a conference”.
- 4: Achievement. Non-cumulative and atomic, like “to break a pane of glass”.

Z-Axis

Instead of using the DOLCE’s definition of TQ, here the values correspond to the definition of the sub-group of TQ named *temporal location*, as proposed by Garbacz. An additional value has been added to the two originals proposed, so they result in:

- 1: Initial SoA. The Behaviour makes the initial PQ or AQ decrease or disappear. Cooling an object makes it lose its initial energy.
- 2: Immutable SoA. The Behaviour does not vary the PQ or the AQ. So, converting energy does not change the energy level.
- 3: Final SoA. The Behaviour makes a final PQ or AQ appear, or increases the existing one. Heating an object makes it obtain a final energy.

Therefore, a Behaviour is now represented as a vector (xi, yj, zk). So, for example, a blowtorch that is used to weld pipes has a Behaviour (2, 3, 3). That is, X = 2 = topological connectedness, due to the fact that welding refers to physical state of connection. Y = 3 = accomplishment. From the definition of P, it can be seen that welding pipes is non-cumulative, because “to weld a pipe” + “to weld a pipe” = “to weld two pipes”, and it is non-atomic, because “to weld a pipe” requires more than an instantaneous action. And finally, Z = 3 = final SoA, because of the fact that topological connectedness is not present at the beginning, but it is obtained at the end.

## 3 DISCUSSION

At this point, it is inevitable to make a comparison between the NIST functional basis and the B-Cube model. The first difference is that, while terms in the functional basis are generic, the ones in the B-Cube are more specific. Hence, when using a tertiary level term from the functional basis, which is the more specific level, like *divide* for example, a lot of information is missing. There is some general information about what the function is doing, i.e. dividing something, but there is no information about what is dividing (a flow is needed), or how the division is achieved. That data helps designers to understand better what is requested by the design or what exactly is being carried out by the device. The simple word *divide* can mean that something material is broken suddenly (2, 4, 1), like a pencil being snapped, or it can mean that something material is being divided into two parts in a less sudden way, like cutting slices of bread with a knife (2, 3, 1), or also a device that splits a signal into two channels (5, 2, 1). The reason for this main difference can easily be found if we remember the FBS framework. Within this framework it can be seen that the functional basis talks about Functions, while B-Cube refers to Behaviours.

This specificity of the B-Cube model, together with the combination of its terms, means that there are a considerable number of terms. The functional basis has seven terms on the primary level, 20 on the secondary level, and 24 on the tertiary one. B-Cube, on the other hand, has at its disposal 96 terms, as a combination of eight X-values, four Y-values, and three Z-values. Moreover, the functional basis is

limited to mechanical design, whereas B-Cube is supposed to cover the full range of design. One important part of this range is the role Behaviours, which are necessary for the process and organizational level design, but are not present within the functional basis.
Another way to explain this difference in level of specificity between the NIST functional basis and the B-Cube model is that, while the former is clearly a taxonomy, B-Cube acts more like an ontology. This seems logical because B-Cube was built from DOLCE's meta-ontology. Being an ontology means there is knowledge hidden inside the term or taxon. Thus, the functional basis is a taxonomy, which means that the term *join* belongs to the group *couple*, which is at the same time a part of the group *connect*.

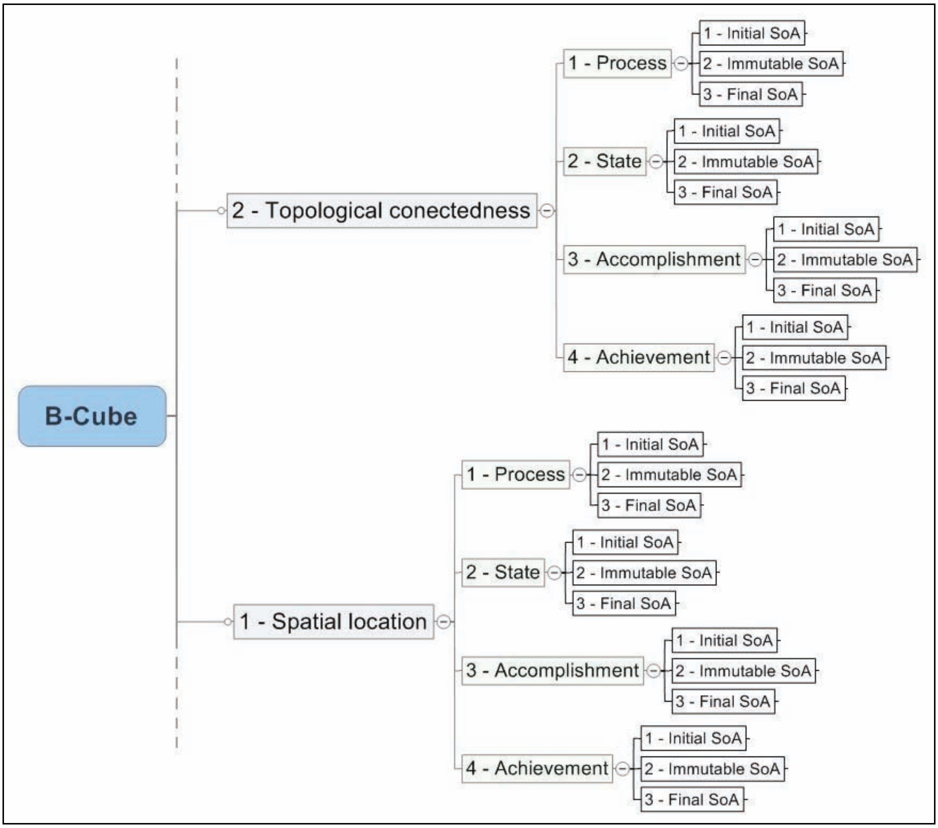


*Figure 2.- B-Cube's ontology*

As can be seen in Figure 2, when we structure the terms hierarchically they are systematically repeated. But this is because it is not a simple classification. Here, each chosen branch provides the final term with a specific meaning. This is one reason for defending the idea that the path is more important than the final term in the B-Cube model. As a result, when we are talking about the path (2, 4, 3), we are referring to an action that affects the topological connectedness (X=2) of an object, in a non-cumulative and atomic way (Y=4), and which makes the object achieve that connectedness (Z=3), regardless of the name we have chosen to designate that path. The terms *touch*, *prick*, *stick*, *attach*, *link*, *join*, and others can fit that meaning, but the same terms can also be understood as another path. For example, *join* can be understood as (2, 3, 3) or (2, 1, 3) as well as (2, 4, 3). Of course a single word can be set for each term, just to make the model easier to understand. But for the time being it seems better to keep the vector terminology in order to avoid the ambiguity that may be present in the interpretation of one simple verb.

## 4 CONCLUSIONS

It has been seen from the comparison between the NIST functional basis and B-Cube model that the latter presents several advantages over the former: a greater number of terms, more specificity in their meanings, hidden knowledge, and it covers the full range of design. Nonetheless, comparing them may not be the best way to deal with them. If we remember the definitions in the FBS framework, the NIST functional basis can be assimilated with the definition of the Function level, and the B-Cube model can do the same with that of the Behaviour level. So in this case, the functional basis and B-Cube are complementary, each at a different level within the FBS framework.

Figure 3 shows an example of how can they work together in functional design modeling. In this example we want to level a surface of a certain material. We can achieve this by polishing the surface (Figure 3a) or by shearing it (Figure 3b). Translated into the functional basis, both of the tools perform the same function, that is, they remove part of the material. But they perform that function in two different ways. In case A, the function is achieved by a process (Y=1) of rotating a sanding wheel, while in case B, it is achieved by the instantaneous action (Y=4) of dropping a sharp blade.

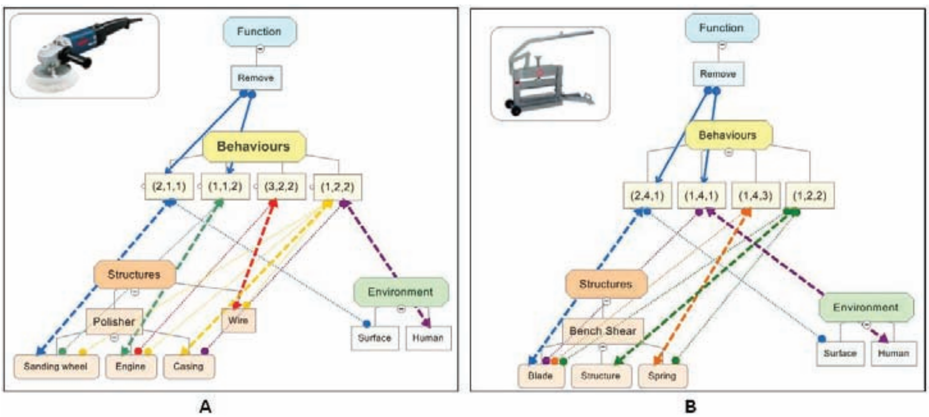


*Figure 3. Example of functional design modeling that combines the functional basis and B-Cube model*

But on the other hand, the NIST functional basis is presented as a taxonomy at the mechanical level only. This means that the functional basis is only useful if designing mechanical tools but, as the B-Cube model is expected to work at all design levels, we will need to widen the function taxonomy.

## ACKNOWLEDGEMENTS

The authors would like to acknowledge funding from the *Ministerio de Educación y Ciencia* of Spain (ref. DPI2006-15570-C02-00) and FEDER funds from the European Union.

## REFERENCES

[1] Gilchrist A. Thesauri, Taxonomies and Ontologies - an Etymological Note. *Journal of Documentation*, 2003, 59 (1).

[2] Gershenson J.A. and Stauffer L.A. The Creation of a Taxonomy for Manufacturability Design Requirements. In *Design Engineering Technical Conferences*, Vol. 2, pp. 305 - 314.

[3] Szykman S., Racz J. and Sriram R. The Representation of Function in Computer-Based Design. In *Design Engineering Technical Conferences*, Las Vegas, Nevada, September 12-15, 1999

[4] Feygenson N. Function Synthesis: New Methodological Tool and Case Studies. In *ETRIA TRIZ Futures*, *Vol.* Kortrijk, Belgium, October 2006

[5] Golden I.J. *Function Based Archival and Retrieval: Developing a Repository of Biologically Inspired Product Concepts*, 2005 (University of Maryland).

[6] Hirtz J., Stone R., McAdams D., Szykman S. and Wood K. A Functional Basis for Engineering Design: Reconciling and Evolving Previous Efforts. *Research in Engineering Design*, 2002, 13 65 - 82.

[7] Pahl G. and Beitz W. *Engineering Design*, 1984 (Design Council).

[8] Hundal M. A Systematic Method for Developing Function Structures, Solutions and Concept Variants. *Mechanism and Machine Theory*, 1990, 25 243-256.

[9] Altshuller G. *Creativity as an Exact Science: The Theory of the Solution of Inventive Problems*, 1984 (Gordon and Breach Science Publishers).

[10] Gero J.S. Design Prototypes: A Knowledge Representation Schema for Design. *AI magazine*, 1990, 11 (4), 26 - 36.

[11] Takeda H., Yoshioka M., Tomiyama T. and Shimomura Y. Analysis of Design Processes by Function, Behavior and Structure. In *The Delft Protocols Workshop, conference proceedings.*

[12] Gero J. and Kannengiesser U. The Situated Function-Behaviour-Structure Framework In (Eds) J. Gero. *Artificial Intelligence in Design 02*, 2002.

[13] Gero J.S. and Kannengiesser U. The Situated Function-Behaviour-Structure Framework. *Design Studies*, 2004, 25 (4), 373-391.

[14] Chandrasekaran B. Representing Function: Relating Functional Representation and Functional Modelling Research Streams. *AIEDAM Artificial Intellig. for Engineering Design*, 2005, 19 (2), 65-74.

[15] Hutcheson R.S., McAdams D.A., Stone R.B. and Tumer4 I.Y. Function-Based Systems Engineering (Fuse). In *16th International Conference on Engineering Design*, *Vol.* Paris, France, 28-30 August.

[16] Kitamura Y. and Mizoguchi R. Ontology-Based Systematization of Functional Knowledge. *Journal of Engineering Design*, 2004, 15 (4), 327-351.

[17] Umeda Y., Kondoh S., Shimodura Y. and Tomiyama T. Development of Design Methodology for Upgradable Products Based on Function-Behaviour-State Modelling. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing (AIEDAM)*, 2005, 19 161-182.

[18] Vermaas P.E. and Dorst K. On the Conceptual Framework of John Gero's Fbs-Model and the Prescriptive Aims of Design Methodology. *Design Studies*, 2007, 28 (2), 133-157.

[19] Xu Q.L., Ong S.K. and Nee A.Y.C. Function-Based Design Synthesis Approach to Design Reuse. *Research in engineering design*, 2006, 17 27-44.

[20] Deng Y. *Functional Design of Mechanical Products: Design Model and Modeling Framework*, 2000 (Nanyang Technological University).

[21] Tor S.B., Britton G.A., Zhang W.Y. and Deng Y.M. Guiding Functional Design of Mechanical Products through Rule-Based Causal Behavioural Reasoning. *International Journal of Production Research*, 2002, 40 (3), 667-682.

[22] Zhang W.Y., Tor S.B., Britton G.A. and Deng Y.M. Functional Design of Mechanical Products Based on Behaviour-Driven Function-Environment-Structure Modelling Framework. 2002, 8.

[23] Rasmussen J. Skills, Rules, and Knowledge; Signals, Signs and Symbols, and Other Distinctions in Human Performance Models. *IEEE Transactions on systems, man, and cybernetics*, 1983, SMC-13 (3), 257 - 266.

[24] Gero J.S., Tham K.W. and Lee H.S. Behaviour: A Link between Function and Structure in Design. In *IFIP - Intelligent Computer Aided Design*, *Vol.* North Holland, pp.193-220.

[25] Borgo S., Gangemi A., Guarino N., Masolo C. and Oltramari A. *Wonderweb Deliverable D15*, 2002 (Laboratory For Applied Ontology - ISTC-CNR).

[26] Borgo S., Carrara M., Vermaas P.E. and Garbacz P. Behaviour of a Technical Artefact: An Ontological Perspective in Engineering. *Frontiers in Artificial Intelligence and Applications*, 2006, 150 (Formal Ontology in Information Systems), 214–225.
[27] Masolo C., Borgo S., Gangemi A., Guarino N. and Oltramari A. *Wonderweb Deliverable D18*, 2003 (Laboratory For Applied Ontology - ISTC-CNR).
[28] Gruber T.R. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. 1993,
[29] Gómez-Pérez A., Fernández-López M. and Corcho O. *Ontological Engineering with Examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web*, 2004.
[30] Chulvi V., Vidal R. and Cebrian-Tarrasón D. B-Cube. In *XII Congreso Internacional de Ingeniería de Proyectos (AEIPRO)*, *Vol.* Zaragoza, 8-11 July.
[31] Chulvi V. and Vidal R. Aplicabilidad de B-Cube al Sector Servicios. In *XII Congreso Internacional de Ingeniería de Proyectos (AEIPRO)*, *Vol.* Zaragoza, 8-11 July.
[32] Garbacz P. Towards a Standard Taxonomy of Artifact Functions. *Applied Ontology*, 2006, 1 (3), 221-236.
[33] Nagel R.L., Bohm M.R., Stone R.B. and McAdams D.A. A Representation of Carrier Flows for Functional Design. In *16th International Conference on Engineering Design*, *Vol.* Paris, France, 28-30 August.

Contact: Vicent Chulvi
University Jaume I
Department of Enginyeria Mecànica I Construcció
Av. Sos Baynat s/n
12071, Castellón (Spain)
Tel. +34 964729252
Fax. +34 964728106
chulvi@emc.uji.es

V. Chulvi is Assistant Professor in the Department of Enginyeria Mecánica i Construcció at the Universitat Jaume I of Castellón (Spain) and researcher at the GID (Engineering Design Group) at the same University. He earned a BsC degree in Mechanical Engineering (2001) and a MsC degree in Mechanical Engineering (2007), and he is currently in the PhD program of Technological Innovation Projects in Product and Process Engineering.

R. Vidal is Professor in Engineering Design. During the last fifteen years she has had different academic positions in the Universitat Jaume I. She is head of GID (Engineering Design Group) and vice-chancellor of the INCREA Chair of Innovation, Creativity and Learning at the same university. She earned a BsC degree in Industrial Chemical Engineering (1990), a MsC degree in Mechanical Engineering (1993), and a PhD in Engineering (1996).