

COMPLEXITY OF PRODUCT DEVELOPMENT CONTEXT

M. ŠTORGA

University of Zagreb, Croatia
Faculty of Mechanical Engineering and Naval Architecture
e-mail: mario.storga@fsb.hr

M. M. ANDREASEN

Technical University of Denmark
Department of Mechanical Engineering
Section of Engineering Design and Product Development
e-mail: mma@mek.dtu.dk

Keywords: product development context, complexity, ontology

***Abstract:** The complexity of design and designing is not something that we can influence on, but the way we model, classify, illustrate and structure our views upon design and designing, strongly influence our perceived complexity. Our research focus is on product development (PD) context as the entire body of data, information and engineering knowledge related to design itself, that evolves throughout the product development effort. The nature of (PD) context complexity is explained by two dimensions: PD context elements description, and PD context evolution. Standardization of the PD ontology is proposed as a formal method for organizing the PD context, in order to improve the robustness and computability of PD context representation and decrease its complexity.*

1. INTRODUCTION

Modern product development is a creative, multidisciplinary process [1]. Accordingly to this definition, the three main disciplines that are involved during product development are: marketing/sales, product development/design and production development/product establishment/ production. The main consequence of those parallel disciplines with business as their common objective is that complexity in modern product development manifests itself in many different ways. Manifestation of complexity depends on designers' consideration of several areas of their work as complex [2, 3]:

- First, the product under development may be considered as complex from both, the constitutive and behavior viewpoint. Complexity of design products results from the combination of many different design disciplines within a single assembly, large number of complex parts in assembly, complex geometry or multiple functions within an individual part, and many properties requested from multiple stakeholders.
- Second, the process of designing may contain many interrelated stages, tasks, activi-

ties and actions. The designers first extend the solution space by diverging from the well-known aspects of design situations while identifying features of the problem which permit a valuable and feasible solution. Creativity, pattern-making, insight and guesswork allow designers to transform the results of the divergent search into patterns that may lead to a single solution. Eventually, the designers must converge to the final design by removing uncertainties and design alternatives.

- Third, organization of designers in project teams integrates complex sets of capabilities and experience, decomposed into teams, working groups and individual assignments.

In our research we have focused on the first aspect of the previously defined design complexity areas. Modern products are developed over a period of time through an extensive development process. The designer must determine customer needs and must have the scientific and engineering knowledge to form these needs into working principles of the designed product. Customer needs are characterized by functional and performance requirements which

are constrained by the operational environment, budgetary limitations, and other restrictions. Often, customer needs are ambiguous and incomplete and they change considerably over time. During product development designers need to transform these ambiguous requirements into a concrete design model while accommodating any changes. The information pattern derived from such process is highly complex and contains many interdependencies. To describe such complexity, design models often contain a vast amount of diverse data and information linked together in a variety of undocumented networks (Figure 1).

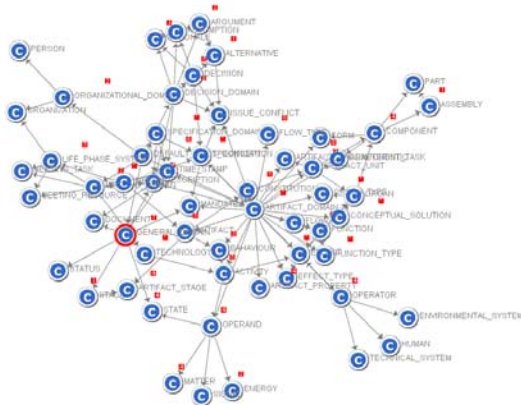


Figure 1. Design model complexity illustrated by functionalities, entities, factors, ... and their relationship.

An important thing is that the complexity of design and designing is not something that we can influence on, but the way we model, classify, illustrate and structure our views upon design and designing, strongly influence our perceived complexity and thereby transparency, operationally, and appropriateness of our operations. In the following we focus upon the possibilities for influencing our perceived complexity. We start with an outline of some observation about complexity of product development context, followed by discussion on methods and techniques for organizing and reducing complexity. Finally we indicate how these methods can help in resolving product development complexity understanding and management problem.

2. THE PRODUCT DEVELOPMENT CONTEXT

As we stated in the introduction chapter, product development is a complex activity that varies widely from one organisation to another reflecting the cultures, working environment and different perceptions of the design teams. Furthermore, researchers in design theory disagree on the nature of development processes. As a result, the modelling of development process and design needs to support a wide range of approaches without imposes limitations on any such approach.

In our research the start viewpoint on the problem is consideration of the product development to be an information process or an information transformation process (following Hubka's definition [4]). The process of transformation from one information state to another is the result of the multiple synthesis and decision making sequences driven by engineering knowledge. The application of knowledge and information is necessary because explicit, limited data that are present as input and output at different stages of product development process is not a sufficient basis for reasonable decision making in engineering design [5].

During product development, designer applies considerable knowledge to the understanding of constitution and behavior of a product design, and forms these cognitions into design models diverse in detail and abstraction level. The engineering knowledge includes different engineering disciplines such as scientific methods, engineering principles, information about existing solutions, standard components, material, manufacturing, etc. In addition, the applicable information frequently changes as the design evolves.

In order to determine the research focus based on the previously described phenomenon, we have defined the *product development (PD) context* as the entire body of data, information and engineering knowledge related to design itself, that evolves throughout the product development effort [6]. Under the vision of a future customizable and flexible product development environment, multiple software tools are used during different product development activities at various stages for creating, adding to, and modifying the elements of the PD context. These tools are incommensurable with existing theoretical models of design and designing and give incomplete data and information, additionally increasing the complexity of product development context. In the next chapters the nature of PD context complexity expansion will be further explained by two dimensions: PD context elements description, and PD context evolution consideration.

2.1. PD context elements

The complex nature of the product development context is hastened by multiple definitions for data, information and knowledge [5] within the different area of engineering design. These many and varied definitions, combined with the fact that data, information and knowledge are often considered to be synonyms of one another, decrease our ability to identify and capture the right piece of information or knowledge that is necessary in particular moment.

About Data

Data is a product of activities as discovery, research, gathering, and creation [5]. Usually data is considered to be textual, either numeric or alphabetical with huge diversity of different formats. But, data is not valuable as engineering communication because

it is not a complete message. Data in engineering sense usually refers to concrete characteristics, especially measured values of specific characteristics of technical system or of other natural phenomenon. That is the reason why data seems to be generally regarded as context free, although this is doubtful [7]. There is no reasonable meaning of data for the “consumers” and too often engineers deluge their team partners with data instead of information, leaving them to sort it out and make sense of it. The consequence of such a situation is increased complexity of the PD context interpretation, caused by different perception of the same data from the particular user viewpoint.

About Information

Information makes data meaningful for users and it requires the creation of relationships and patterns between data. Transforming data into information can be accomplished by organizing it into a meaningful form, presenting it in meaningful and appropriate ways and communicating the context around it. Unfortunately, the design models that engineers are using today provide little or nothing of the meaning for relationship and patterns established between different information. According to the literature, there are two classes of information: formal and informal [5, 8, 9]. The primary difference between informal and formal information is the structured nature of formal information. Information act as an operand of an information transformation process and the variety of its content and structure, its form (verbal, graphical, symbolic), its location and time dimension, cause the increased complexity in information resources during product development.

The Knowledge Experience

With every experience, we acquire knowledge; i.e. understanding gained through experiences – good or bad. Knowledge is communicated by building compelling interactions with others or with tools so that the patterns and meanings in their information can be learned by others. Knowledge is intended to imply information that has been processed in some way obtaining an “accepted true belief” on the basis of evidence or even lack of it. The danger of such an approach is the inherent viewpoint on domain of discourse. Often engineers, doesn’t consider in details received information, they doesn’t try to understand the context of information, and they use information in the way learned from the others in a working environment. To avoid such misunderstandings, knowledge should be derived from information by processes of deduction, induction, reduction/abduction, or innoduction [3, 9]. Some knowledge is personal, having meaning unique to one person’s experiences, thoughts, or point of view. Local knowledge is knowledge shared by few people, based upon their shared experiences. In opposite, global knowledge should be more general, limited and process-based, since it relies on shared understandings and agreements about a domain.

In engineering processes, the knowledge is used as basis for decision-making purposes. If informal information is used to infer knowledge for decision making processes, then the validity and reliability of the knowledge cannot be guaranteed. In order to ensure a high level of confidence in the inferred knowledge it must be based on clearly defined facts (formal information), not on personal interpretation of facts or inherent believes as in the case of informal information [5]. Today’s information technology can easily capture, transform, and distribute large amounts of highly structured knowledge. But, for tacit, hard-to-formalize knowledge that must be interpreted in a broader context and combined with other type of information, the human brain is still the best technology. However, information technology should increase the quality of person’s decision making and problem solving, by reducing knowledge complexity and providing relevant informal and formal knowledge in the actual work context.

2.2. PD context evolution

The another dimension for explaining the complex nature of the PD context results from the fact that the PD context evolution during product development operates in two intertwined modes, iterative and layered [10]. The iterative mode accounts for the various feedback loops that occur as designers seek to satisfy design goals. Furthermore, designers develop design solutions by reasoning about the problem at various levels of abstraction (Figure 2) [1, 11].

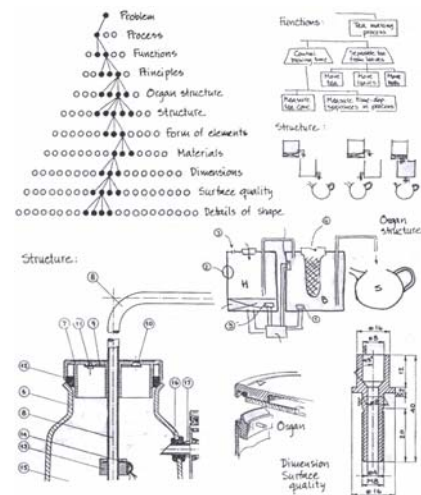


Figure 2. Different abstraction level of design problem [1]

The complicated relationships and large quantities of information and knowledge in a complex design are very difficult for designers to assimilate in their totality. The relationships between designs’ characteristic are often unknown or poorly understood which further exacerbates the problem. In an attempt to manage design complexity, designers usually employ a number of common design activities to decompose the problem into more manageable pieces. The domain oriented approach [11] in the

design modeling corresponds to the abstraction of the design from several viewpoints. A level of abstraction is a view of a design problem that includes only the issue designers are considering relevant from a given viewpoint in the design process, reducing the complexity of the whole problem to the complexity of the actual designer's perception on the partial problem. Designers continuously shift between iterative and layered modes and accumulate information and knowledge generated at various levels of abstraction.

Figure 3 gives a conceptual picture of the relation between product development process and PD context complexity. The development process starts on the bottom of the cone and proceeds with an increasing degree of concretization, through areas associated with market, design and production [1]. At the beginning of the development process, designers are uncertain about the details of design parameters and characteristics. All these uncertainties increase the numbers and combinations of possible outcomes. The outer spiral represents product development context. The spiral widens from bottom to top, indicating that the amount and complexity of information and knowledge is increasing as the design approaches completion.

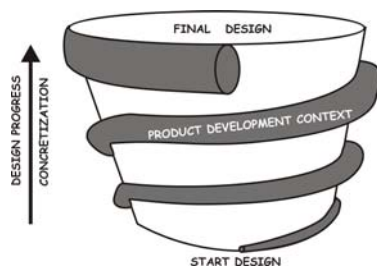


Figure 3. *PD context evolution*

3. ORGANIZING AND REDUCING THE PD CONTEXT COMPLEXITY

It is recognized for a while that insight into PD context is one of an enterprise's most important assets, decisively influencing its competitiveness. Large engineering projects involve the resources of many different clusters of cooperative subjects (human and computer) in the given situation. Each cluster makes its own contributions, and the overall success of the project depends in large measure on the degree of integration between those different clusters throughout the development process. In addition to the dynamic and complex nature of the PD context, an enormous problem in the coordination of large projects is the diversity of backgrounds the various kinds of engineers bring to their respective role. As a consequence, many engineers use similar terminology for describing the PD context elements, in many different ways with many different connotations increasing on such way the complexity of the PD context. Because of such differences, the informa-

tion and knowledge that one engineer intends to convey to another may in fact become garbled; in the best case such miscommunications can be responsible for a great deal of lost time and resources; in the worst can result in the lost of life. To avoid this situation, in our research we believe that is necessary to unify the vocabulary for spelling the PD context and define its meaning in the appropriate situation of use.

Any domain with a determinate subject matter has its own terminology, a distinctive vocabulary that is used when talking about characteristic objects and processes that compromise domain. But the nature of the domain is not revealed in its corresponding vocabulary alone. In addition, rigorous definition of the rules governing the way terms in vocabulary can be combined to form the statements should be provided and the logical connections between such statements should be clarified. Only when this additional information is available, it became possible to understand both the natures of the individual concepts that exist in the domain and the critical relationships they bear to one another [12].

An ontology is a structured representation of such information. More exactly, an ontology is a domain vocabulary together with set of precise definitions, or axioms, that constrain the meanings of the concepts in that vocabulary sufficient to enable consistent interpretation of statements that use that vocabulary. As an objective of our research, we believe that standardization of the product development ontology [6] can be used for formal organizing of the PD context, in order to improve the robustness and computability of PD context representation, to decrease its complexity, and to avoid misinterpretation of it. Further, by the capture of consensual data, information and knowledge in a generic and formal way, PD context may be meaningful reused and shared across different applications (software) and by multiple groups of people.

3.1. Content of the product development ontology

A mixed approach of existing methodologies for building ontologies [6] together with review of the current and past research of product development related topics, have been aimed in our research to successful abstraction and formalization of entities (objects, relationships, rules, attributes, etc.) that are common across the greater part of the product development activities. As a theoretical background for extracting the content of the PD domain, the Genetic Design Model System was chosen [13]. Accordingly to research results, GDMS seems to be able to capture the totality of results created in product development project, and it is a more comprehensive than other design model systems that can be found in literature.

The results of the GDMS research project are in literature presented as proposal of the genetic design language contemplated as the set of the infinite de-

signs, which can be synthesized, based on a design vocabulary and syntactical rules [13]. The principal contents of PD context have been described by three domain languages: function-, organ-, and part language. Each of the languages points out the concepts of different types which can be utilized for creating the design models. Within each domain important question was addressed during our research: what basic or core concepts are required to describe situation in particular domain? As the result of the analyze core of about 100 different concepts was extracted, following the directions of the GDMS authors to concepts that should be used as the elements of the design vocabulary. Extracted concepts are of wide variety e.g. activities, arguments, assemblies, components, conditions, criteria, decisions, dimensions, effects, features, transformations, forms, functions, materials, needs, operands, organs, parts, revisions, states, technology, work subjects, etc., but in this article is not a enough space to quote all of them.

In order to formally define the meaning of every extracted concept our presumption was that the meaning of every particular concept can be formally defined by means of attributes (data) related to the concept and by relations with other concepts. In other words, each definition of concepts in PD ontology requires careful understanding of it in relationship to the other concepts in the ontology. The first proposal for an ontology, after extracting the core concepts had an informal form, consisting of concepts and definitions shown in natural language. The concepts have been chosen according to the theoretical foundations as far as possible to match the natural use of English word by people participating in product development. The few examples of extracted informal definitions are:

- TRANSFORMATION describes PRODUCT BEHAVIOUR.
- TRANSFORMATION is activity within LIFE-CYCLE MEETING.
- TRANSFORMATION has as input an OPERAND in a STATE.
- OPERATOR is agent of activity TRANSFORMATION.
- OPERATION follow (proceed) OPERATION.
- ENVIRONMENT SYSTEM is kind of OPERAND.
- ORGAN is realized by COMPONENT
- PART is part of ASSEMBLY
- ...

During the whole concept extraction phase, the hundreds of similar definitions were derived, defining the relationship between concepts in the same domain as well as between concepts in different domains. For every particular concept five to ten key definitions was stated/formulated, in order to determine the position of the concept in PD vocabulary. Informal definitions displayed another problem and new research questions arise at this point of the research. Every definition contents the relation beside the concepts e.g. *is result, follow, describe, is kind of, has an input*, etc. The consequence is a huge diversity of relations and for most of them there does not exists explanation of meaning in the background

theory. Most of them are in the theory characterized as causal, only to denote their existence, without further explanation of their nature. After consideration of the huge number of unclassified and undefined relations that makes the semantic network between concepts in PD ontology complex, we highlighted it as the one of the biggest obstacle in fully formalization of PD context.

In order to formalize the meaning of the different relations, the attributes and axioms for relations should be defined. The first step to solving this problem was a classification of the different relationships by their nature and characterization of commonly used relations that exists between concepts in PD domain.

3.2 Nature of the relations between concepts

The motivation for further research on relations between concepts in PD domain, should be the possibility of creating the repository of the formally defined and characterized relations that can be reused and customized all along the product development process. The same or similar relations will likely appear in a number of different ontologies for different domains during the product development. The standards and literature provide little guidance and do not provide detailed guidelines on what semantic relation appear in design models [14, 15]. Following the approaches in literature, the relations between concepts can be grouped into the following categories:

1. Classification relations (taxonomy) – capture semantic of *kinds* and *types*;
2. Meronymic relations (mareology) – capture semantic of *whole/part* concept;
3. Temporal relations (temporal logic) - capture semantic of the *time depend* relations;
4. Connectivity relations (topology) – capture semantic of the *geometric, physical and other form of connections, contacts or interactions*;
5. Dependency relations – to express that an object *depends* on another
6. Influence relations - to express that an object has some *effect* or *impact* on another object;
7. Other relations - they don't depend only on the nature or the meaning of the terms they relate but also should provide the knowledge behind structure.

We believe that every relation that we separated during informal definition of PD ontology can be classified in one of the previously defined group. The properties of the every particular relation can be derived from the properties of the group and it can be the step closer to reducing the complexity of the PD context semantic network.

4. CONCLUSION

It is obviously from the previous discussion that complexity of the product development context is a

multidimensional problem growing in complexity with context evolution during the development process.

In order to reduce complexity and better organize the PD context, we propose the effort aimed to the unifying of the product development ontology. Is this a feasible, powerful way to cope with complexity in design? We should consider the wider picture in order to get the affirmative answer on this question.

In this picture the irrefutable fact is that we cannot avoid "spelling about design". We usually spell the design by using the natural language, but on such way we loose possibilities to control and rule description of the situation in domain of discourse. The point of our proposal is in using the formal language with precisely defined rules to resolve the complex principles behind design as a phenomenon.

The concepts of the formal language are defined based upon the theoretical models of design and designing. The problem that appears in this approach is that background theories provide insufficient directives about the nature and meaning of the relations between concepts. We believe that the first step to solve this problem can be research on the semantic relations between concepts. This approach includes definition and indication of how concepts are inter-related and constrains on the possible interpretations of concepts. The characterization of the relation groups that we proposed, should be in a future research performed by defining the nature of relations using the logical viewpoint in order to enable performing the more "intelligent" treatment on these relations.

Concerning the practical role of PD ontology approach, we believe that a key aspect is its capability to explicate tacit knowledge required for the real-life tasks and knowledge structured in accordance with the theoretical design models. The main benefits of the learning efforts needed to use PD ontology, is in fact that in this way the designers can enhance PD context reuse and provide efficient learning support for end users that are less experienced.

Acknowledgements: This research is part of funded project "Models and methods of improving the computer aided product development" supported by the Ministry of Science and Technology of the Republic of Croatia. The presented paper is result of the research performed in co-operation with the scientific staff of Section of Engineering Design and Product Development, Technical University of Denmark.

References

- [1] Andreasen, M.M., Hein, L.: *"Integrated product development"*; IPD; Lyngby; Denmark; 2000.
- [2] Eppinger, S.D.: *"Patterns of Product Development Interactions"*; Proceedings of the International Conference on Engineering Design ICED 01, Glasgow, Scotland, 2001.
- [3] Earl, C., Eckert, C., Johnson, J.: *"Complexity models in design"*; Proceedings of the 8th international design conference DESIGN 2004; Dubrovnik, Croatia; Marjanović, D. (ed.); FSB Zagreb; The Design Society Glasgow; pp. 163-168; 2004.
- [4] Hubka, V.: *"Practical studies in systematic design"*; ButterWorth Scientific Co.; UK; 1988.
- [5] Hicks, B.J., Culley, S.J., Allen, R.D., Mullineux, G.: *"A framework for the requirements of capturing, storing and reusing information and knowledge in engineering design"*; International Journal of Information Management 22; Pergamon; pp. 263-280; 2002.
- [6] Štorga, M., Marjanović, D., Bojčetić, N.: *"Consideration on IT systems interoperability in product development"*, Proceedings of the TMCE 2004, Millpress, Rotterdam, ISBN, 2004.
- [7] Eder, W.E.: *"Information – a taxonomy and interpretation"*; Proceedings of the 8th international design conference DESIGN 2004; Dubrovnik, Croatia; Marjanović, D. (ed.); FSB Zagreb; The Design Society Glasgow; pp. 169-176; 2004.
- [8] McMahon, C.A., Pitt, D.J., Yang, Y., Sims Williams, J.H.: *"An information management system for informal design data"*; Engineering with computers, 11; pp. 123-135; 1995.
- [9] Abecker, A., Bernardi, A., Hinkelmann, K., Kuhn, O., Sintek, M.: *"Toward a Technology for Organizational Memories"*, IEEE Intelligent systems, 1998.
- [10] Shooter, S.B., Keirouz, W.T., Szykman, S., Fenves, S.J.: *"A model of flow of design information in product development"*; Engineering with Computers; Springer-Verlag London; 1999.
- [11] Mortensen, N.H., Andreasen, M.M.: *"Designing in an interplay with a product model"*; V.96-03; 1996.
- [12] Borst, W.N.: *"Construction of engineering ontologies for knowledge sharing and reuse"*; CTIT Ph. D-Thesis series No. 97-14, Centre for Telematics and Information Technology, The Netherlands; 1997.
- [13] Mortensen, N.H.: *"Design modelling in a Designer's Workbench – Contribution to a Design Language"*; PhD thesis; IKS 00.02.A; DTU; 1999.
- [14] Knowledge Base Systems: *"IDEF 05 Method Report"*; 1994.
- [15] Salustri, F.A., Lockledge, J.C.: *"Towards a formal theory of products including mereology"*; Proceedings of the International Conference on Engineering Design ICED 99, Munich, Germany, 1999.