Conceptual Models of Information Content for Product Modeling

Yatish Bathla

Óbuda University, Doctoral School of Applied Informatics and Applied
Mathematics, Bécsi út 96/b, H-1034 Budapest, Hungary
e-mail: yatish.bathla@phd.uni-obuda.hu

Abstract: Modeling of product as a system is the new era of virtual product development. Requirement, Functional, Logical, and Physical (RFLP) structure provides system engineering based product definition in product modeling which allows to more effectively translate requirements of the customers into the physical form. But, it is difficult in RFLP structure to store and organize the knowledge of product model in a generic format so that engineers of different disciplines gain detailed knowledge of participating components in order to make the most effective decision. In this paper, a conceptual model is proposed by classifying the information content. This is done in context of the engineering discipline and system behavior. Info-Chunk is proposed in the logical level of RFLP structure. They are mapped with the information content and describe the parameters of the model for representing the engineering object. The models are used to guide the engineer to a precise correlating decision.

Keywords: RFLP structure; Product modeling procedures; Information content of product model; Engineering objects management; Life cycle management of product information; Engineering discipline; System behavior

1 Introduction

A complex product model generally faces difficulties in making decisions when there is a large number of engineering objects participating in the product modeling. The level of complexity increases in the case of high number of dependencies among the engineering objects. In a complex multi-discipline product, the difficulty level is higher because there is a vast collection of data gathered from the various engineering disciplines participating in product modeling. In the virtual environment, some of the challenging tasks need definite and correlated information of an engineering discipline, tracking activities of the system behavior, amongst others. Hence, the establishment of effective assistance of engineering decisions is quite impractical. To cope with the issue, information content was defined to control the engineering object and related engineering activities. However, the information content is in the unfledged stage and needs to be enhanced in terms of human-computer interaction and structured processing of interrelated engineering objects to obtain coordinated decisions. This paper proposes classification of information content by classifying the intent so that the decision processes can take place efficiently. These are discipline based content and behavior based content respectively. Based on the classification, a different number of conceptual models are generated to store the knowledge of engineering discipline and system behavior. This knowledge is stored and applied to take effective decisions so that the complex product model is represented in a simplified manner. Info-Chunk is introduced in the logical component and logical level of RFLP (requirement, functional, logical and physical) structure. This entity is mapped with the information content to control the activities of the product model. In the present situation, RFLP structure requires a sector that can store and organize knowledge of the product model in generic format for the engineering discipline to gain knowledge of product model and provide the guidance toward the most effective decision. Community diagrams [10] are used to represent the parameters of information content that provide assistance to the engineering discipline and system behavior. The rest of the paper is discussed as follows: It starts with the preliminary work. Information content is then classified by engineering discipline and system behavior. After the classification, Info-Chunk is defined for the conceptual model. Later, the configuration of info chunk in the logical component and logical level is explained which drives the conceptual model. Next, the decision strategy of the conceptual models are discussed. After that, a practical approach of conceptual model in the real world is explained. Finally, the conclusion and future work are discussed.

2 Preliminaries and Purpose

Classical Product Model (CPM) [13] represents engineering objects with certain types of attributes and their relationships. CPM, however, doesn’t provide sufficient information to analyze or upgrade the current product model. To fill the gap between the engineer and information based product modeling procedures, information content [1] was proposed. It is an interactive media to transfer the content information from human to data based computers. It is suitable for better explanation and evaluation for the making of an interrelated decision on product objects [7]. Here, the term *information content* is used related to the technical process of the system. To calculate an engineering object related data, information content sector is interconnected with data oriented product model sector. The data oriented sector consists of the engineering objects’ description, their attributes and information about their functions and activities. In other words, knowledge of the engineering object is stored in the data-oriented sector. In the context of this research, a mathematical model is used in the RFLP structure to obtain the information about a system. Artifacts represented by the mathematical models are called virtual prototypes [3] that can optimize the product properties without a physical prototype. Info-Chunk is an entity [2] that transfers the knowledge of the product model to the conceptual model of Information content. Likewise, there are different kinds of entities used in the conceptual model. It can be based on type, shape, size or property of a system. Engineering objects and entities define a product in the virtual environment. These terms are related to each other but not the same. High-level entities use low-level entities as a parameter of model creating procedures. It could be an engineering object [15], a component [10] or a process [6]. Information content is a sector, which is used to control the system in the virtual environment. The controlling data can be input manually or automatically. In the case of manual information content, the engineer can initialize the parameter of info chunks using the specialized knowledge. Whereas, in case of self-adaptive information content, the parameter of Info-Chunk is initialized automatically by the intelligent space called Intelligent Virtual Product Space (IVPS) [3] where the development sector, interface sector, behavior sector and learning sector are used to store the knowledge of the system.

3 Current Practice

To analyze the behavior of a product, RFLP structure considers the product as a system [9]. It can consider all aspects of the virtual prototype of a product before manufacturing the first physical prototype. It is a framework that supports the MBSE (Model Based Systems Engineering) process [10]. There are numerous companies investing in the product modeling like Dassault Systèmes, Synopsys, Autodesk, Siemens amongst others. In the context of research, the author considers RFLP structure of CATIA V6 and 3DEXPERIENCE (3DXP) platform by Dassault Systèmes. In this software, Dymola [14] is used to analyze the dynamic logical behavior of a product and Modelica [5] is used for logical and physical modeling of the technical system. Modelica is a multi-domain modeling language for component-oriented modeling of complex systems and based on the object-oriented principles. To generate the executable code, Modelica uses Dymola compiler, which generates C code [12]. This code can run on the hardware and can possibly be modified manually to implement the interfaces of controller API.

4 Classification of Information Content

The role of information content in the product model is to store the relevant information of a system. Information must be pertinent and precise so that the engineers can understand every aspect of a system and take the literal decision. The conceptual model of information content can control the product model by accessing the parameters of Info-Chunk in the RFLP structure. Classification of information content is based on the engineering discipline and system behavior by using the information stored in the info chunk. It is categorised as discipline-based content and behavior-based content as shown in Fig. 1 respectively [1]. Discipline based content stores the knowledge of various engineering disciplines while behavior based content stores knowledge of the system behavior. It can also be used for upgrading the product version by adding new features, parts or subsystems.

The RFLP structure is compliant with the IEEE1220 standard. It is based on the V-cycle design process and allows concurrent engineering to coordinate the separate activities of a distributed design team. The conceptual models of Information content are mapped with logical and physical levels of the RFLP structure as shown in Fig. 2 [11]. Here, both contents are inter-connected with each other so that any changes made in content affects other content. Furthermore, a different number of models can be constructed for a system based on the classification. To explain the concept of the content, let's consider: number of disciplines in discipline based content = N_d, number of behaviors in behavior based content = N_b, number of disciplines participated in the engineering activities = D, number of expected behaviors of a system = B.

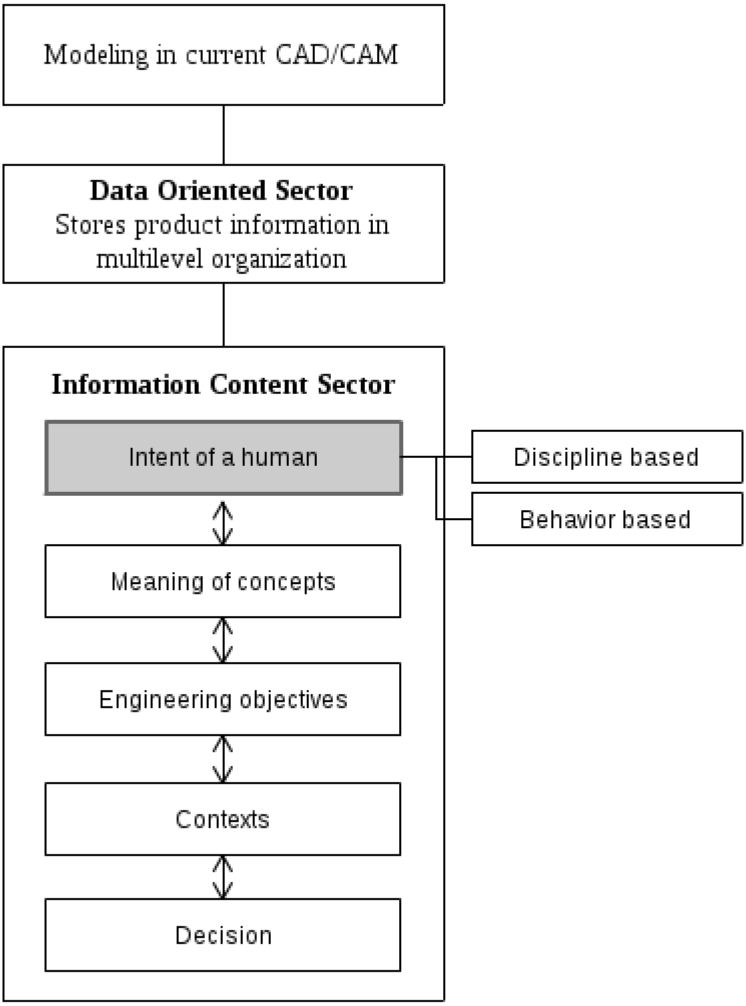


Figure 1
Category of content in the information content sector

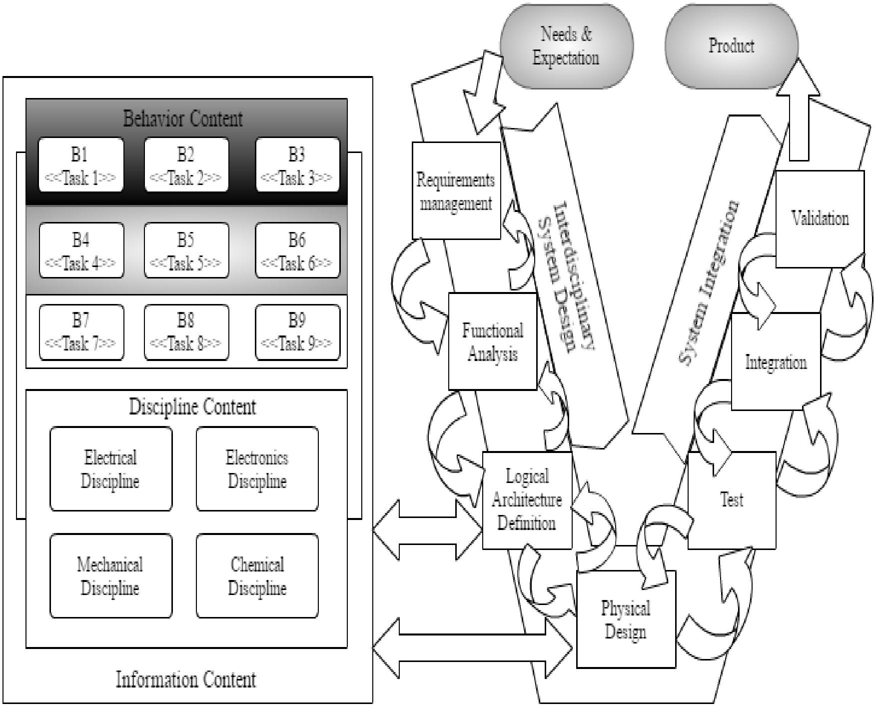


Figure 2

Relation between proposed information content and RFLP structure

4.1 Behavior based Content

The intent of behavior-based content is defined based on system behavior. The engineering objective is to view the system within the context of expected behavior. Behavior based content is the collection of expected behaviors based on the requirement. Priority [8] is assigned to a behavior in the content and arranged in descending order as shown in Fig. 2. In terms of expression, $N_b <= B$, as some behaviors of a product are more mandatory to implement than others in the real world. Here, low priority behavior could be implemented in a future version of the system.

4.2 Discipline based Content

Discipline based content is defined by intializing the intent based on the number of participating engineering disciplines within the system. The engineering objective is to view the system in the context of a discipline as shown in Fig. 2.

Engineers can store the knowledge of a discipline in the Info-Chunk which will be discussed in the next section. This information is used by various engineering disciplines for the depth of knowledge of a system. This content is used to display the activity of a discipline in the product model. In terms of expression, $N_d = D$.

5 Info-Chunk

Info-Chunk acts as a tunnel between information content and the RFLP structure. It stores the correlated knowledge of logical and physical components that participate in the product modeling. It is defined manually by the engineer or automatically by the virtual space during the system design phase. Info-Chunk is placed in the logical component and logical layer of RFLP structure. The knowledge of a system is distributed in the small portions of information in the form of an Info-Chunk in RFLP structure and stored in the data-oriented sector. The main goal of Info-Chunk is to deliver the RFLP structure's component knowledge to the Information Content in context of engineering discipline and system behavior. It is categorised into Component Info-Chunk and Layer Info-Chunk explained in the following subsections.

5.1 Component Info-Chunk

Component Info-Chunk (CiC) is a low-level entity and placed in the logical component of the RFLP structure. It can store the knowledge of the logical component. The parameters that describe the CiC are demonstrated in Fig. 3. In Catia V6, CiC is used to store the knowledge of a modelica component based on the configuration. According to the proposed rule, if there is one CiC, then the component name is the modelica component name defined in the library otherwise initialized either by an engineer or by the intelligent virtual space. Here, the component description parameter is optional and defined to store the component name and component number in the case of more than one CiC. The LiC description stores the information of Logical Info-Chunk. It will be described briefly in the next subsection. The community name stores the engineering discipline name of the modelica component. It is the main parameter of discipline based content. The contribution in the product parameter stores the role of Component Info-Chunk in Layer Info-Chunk for the expected result. The connector parameter stores the knowledge of the inner connector and the stream connector. Here, the inner connector is concerned with the knowledge of input port and output port type while the stream connector is concerned with knowledge of the material flow in the component as explained in [5]. The behavior parameter stores the role of component behavior contributing to a behavior of system. It is the main parameter for the behavior based content.

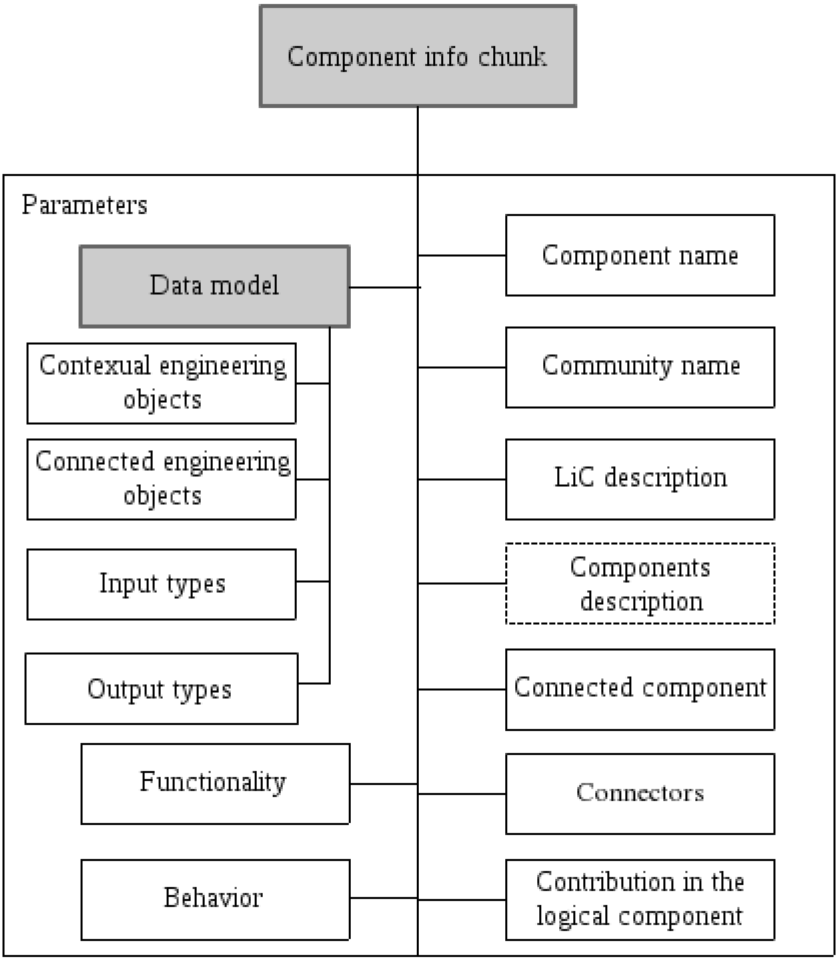


Figure 3
Parameter of Component Info-Chunk

The functionality parameter stores the feature of the component. The data model stores knowledge of the contextual engineering object for the physical level of a system. The contextual engineering object parameter stores knowledge of the engineering object within context of influenced engineering objects and relates to the component. The connected engineering object parameter stores the knowledge of engineering object in the context of connected engineering objects.

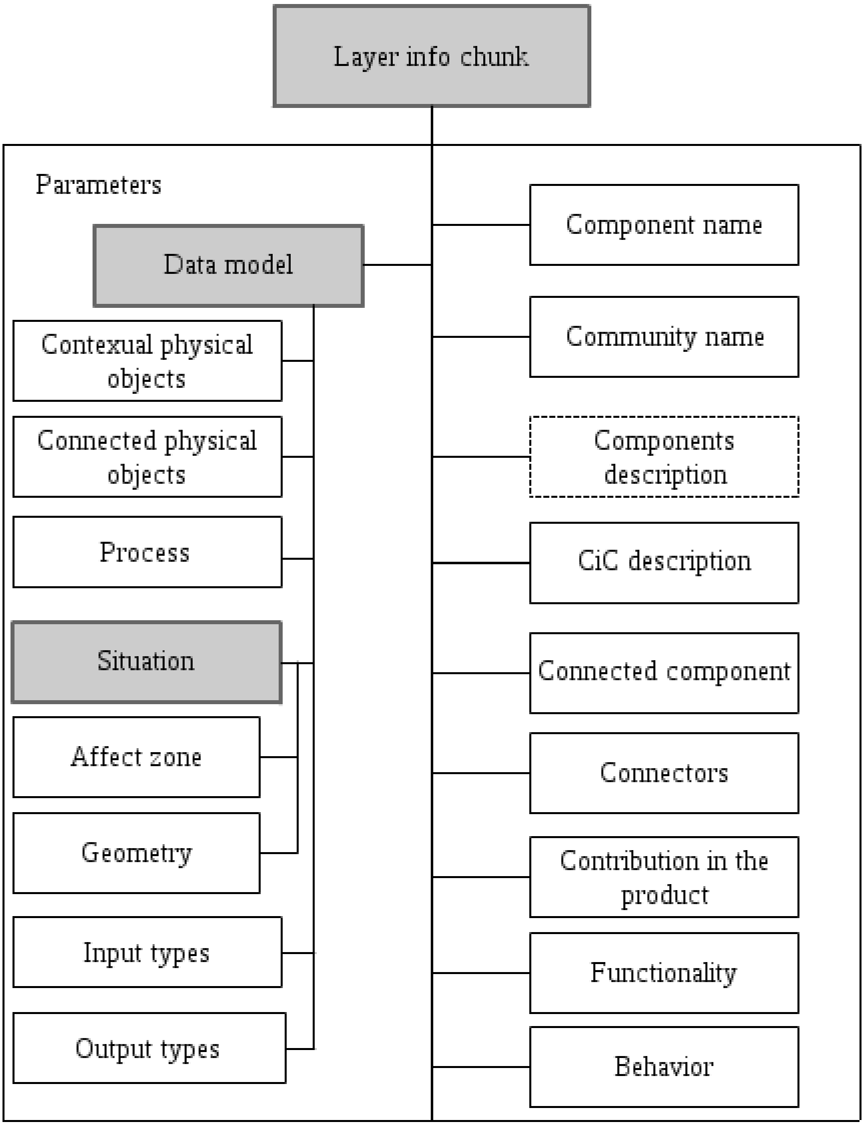


Figure 4
Parameter of Layer Info-Chunk

The input and output type parameters store the knowledge of connection type for input and output port. It depends on the discipline of the connected engineering objects for example if the connected engineering object at the input is a mechanical discipline then the input type parameter will be mechanical. Similarly, the output type parameter will be calculated.

5.2 Layer Info Chunk

Layer Info-Chunk (LiC) is a high-level entity and is placed in logical level of the RFLP structure. The parameters of the LiC are described in Fig. 4. The component name parameter stores the name of the engineering object. Like CiC, the parameters of LiC such as community name, component description, functionality and behavior follow the same steps.

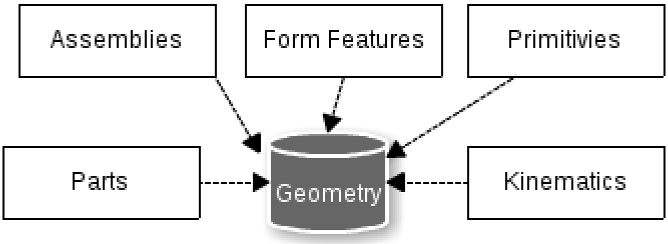


Figure 5
Elements in Geometry entity

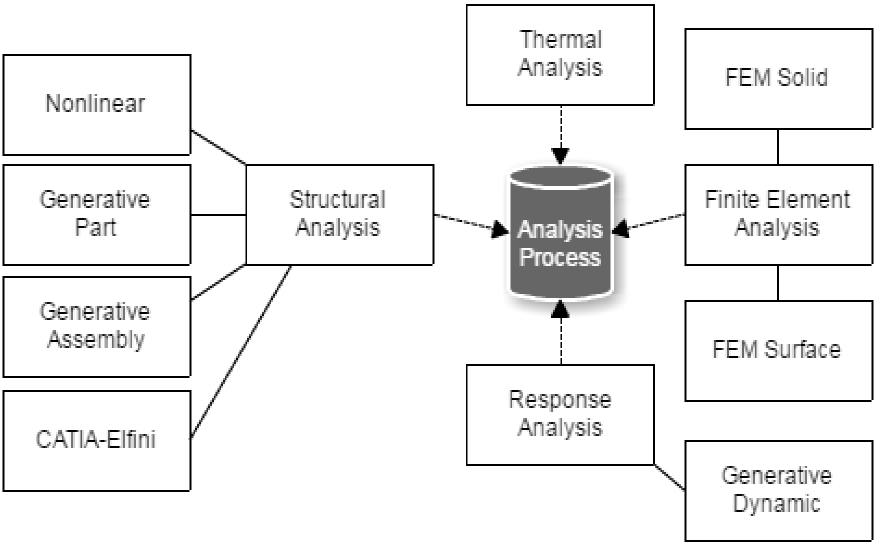


Figure 6
Elements in Analysis Process entity

The connector parameter contains the knowledge of the inner connector and extended connector. The inner connector parameter stores information of the input port and output port type while the extended connector parameter stores information of input and output type of LiC.

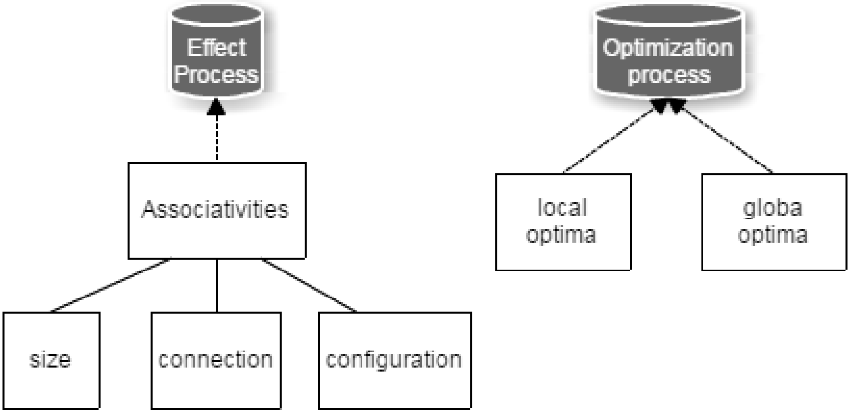
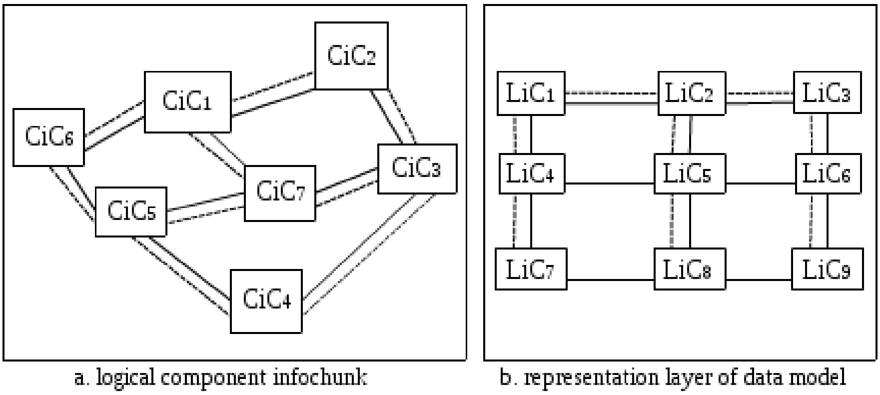


Figure 7
Elements in Effect Process and Optimization Process entity

In case of discipline content, the LiC connector type refers to a discipline while in the case of behavior content, the LiC connector type refers to a behavior. The contribution in the product parameter describes the role of Layer Info-Chunk in logical level of the RFLP structure to deliver the expected result of a system.



a. logical component infochunk b. representation layer of data model

Figure 8
Configuration of Info-Chunk

The data model parameter stores detailed description of the engineering object. The affect zone parameter stores the information of the engineering objects that are influenced due to changes which took place in the analyzed the engineering object. The geometry parameter stores knowledge of the engineering object shape in a situation. It is considered an low-level entity that stores the information of elements like parts, assemblies, form features, and others in this category. The element present in the geometry entity can be assumed as a lower level entity as shown in Fig. 5. The process parameter stores the information of the process involved in the product modeling. As explained in the paper [6], processes involved in the information content are the analysis process, effect process and optimization process respectively. Like geometry, the process is considered as a low-level entity, which stores the information of the process required by the system design phase. This is explained in Figs. 6 & 7. The input type and output type parameters store the i/o connection of the contextual engineering object.

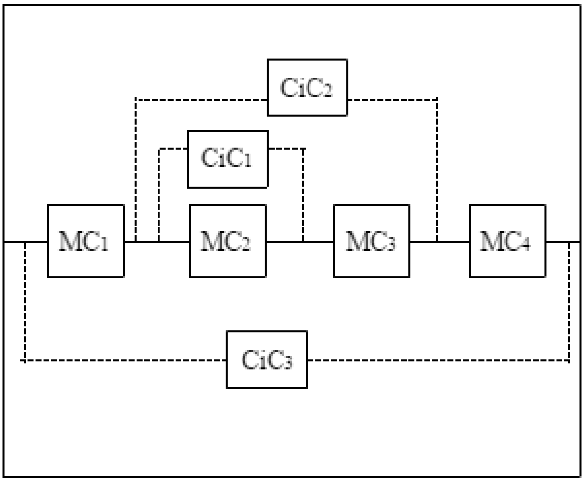
6 Configuration for Driving the Model

Info-Chunk is placed in the logical component or the logical level of the RFLP structure depending on the type. LiC is collection of the CiC and is connected by the logical and physical connection. The configuration of the LiC is shown in Fig. 8a. The logical connection is the connection between the logical components and is demonstrated by a straight line whereas the physical connection is the connection between the physical components and is demonstrated by a dashed line. Furthermore, structure of the LiC depends on the type of information content. In a similar way, the rest of the LiCs are defined during the product modeling. In order to access the information of a system, LiC are arranged in the representation layer of the data-oriented sector by physical connection and logical connection as shown in Fig. 8b. The CiC is placed in the logical component whereas the LiC is placed at the logical level of the RFLP structure. In Catia V6, the CiC is a low-level entity that extracts information and is represented either corresponding to a modelica component (MC) or to a group of modelica components or the entire graph of a logical component as shown in the Fig. 9a. Engineers can initialize the Info-Chunk as per the system specification. Like CiC, the LiC is represented in the logical component (LC) by same steps shown in Fig. 9b. The logical level of the RFLP structure is mapped to data-oriented sector by the Layer Info-Chunk (LiC) which can transfer the product related knowledge to the information content. The data-oriented sector is connected with the information content to take the correlating decisions during the product modeling.

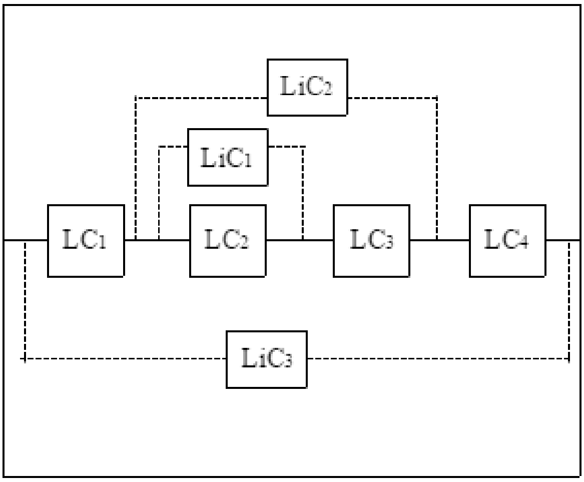
7 Conceptual Models of Information Content

After evaluating the parameters of Info-Chunk, the final step is to construct conceptual models of information content based on the system behavior and engineering discipline. All the product related decision take place in decision level of the information content. It is important to note that data is accessible in the information content. In other words, it is not possible to make any changes in the data-oriented sector directly. Hence, the interface for HCI (Human Computer

Interaction) is required to access the conceptual model of information content and Info-Chunk. The strategy for the conceptual model of behavior content is concerned with the system behavior while the discipline content is concerned with the engineering discipline.



a. logical component



b. logical layer

Figure 9
Representation of Info-Chunk

7.1 Behavior-based Content

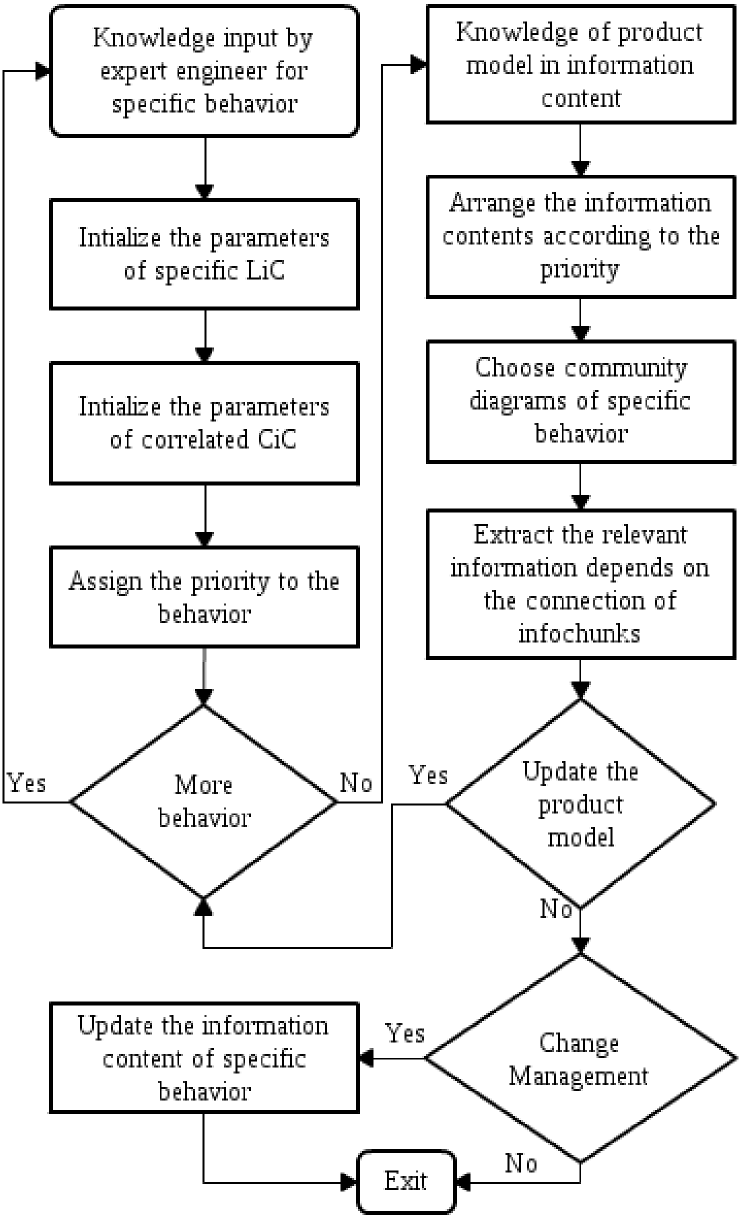


Figure 10
Flow chart of behavior based content

Behavior content is focused on the customer demands. Priority is assigned to the system behavior based on the requirements of customer. Here, some behaviors are more important than others. Therefore, a number of behavior created in a content is based on the priority given to a behavior as explained in Fig. 10. The challenging task is to represent a behavior of a system. It is evaluated and constructed by extracting the mapped parameter of the Component Info-Chunk and Layer Info-Chunk respectively. The logical component with the contextual engineering object of a behavior are initialized. Then, Layer Info-Chunk is proposed and initialized. After that, Component Info-Chunk is proposed and initialized over the Layer Info-Chunk. In a similar way, Layer Info-Chunk is proposed and initialized for the rest of the system behavior. The decision process starts with the community diagram.

To extract the information of a behavior, the community diagram [4] is generated based on the type of Info-Chunk and connection. There are four possible types of community diagrams generated for the behavior content as shown in Fig. 11.

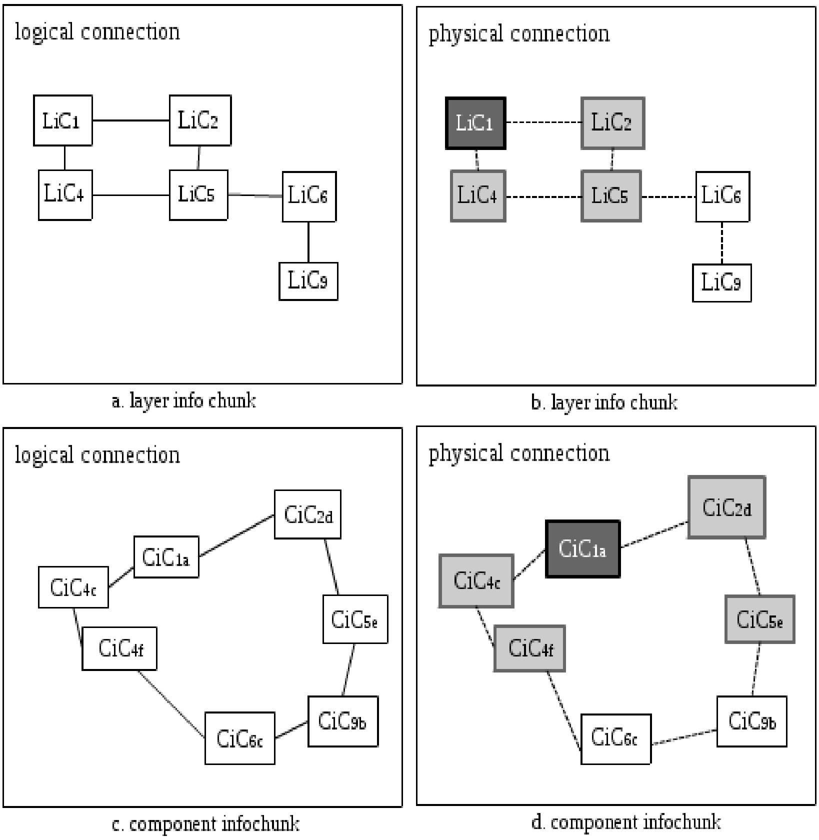


Figure 11

Community diagrams of information content

To express the relation between Info-Chunk in a community diagram, let us assume the following terms: Layer Info-Chunk is represented as LiC, such that $\{LiC_1, LiC_2, LiC_3, \ldots\ldots LiC_m\}$ are the numbers of Layer Info-Chunk present in a logical layer, where m is the total number. Similarly, Component Info-Chunk is represented as CiC, such that $CiC_x = \{CiC_{xa}, CiC_{xb}, CiC_{xc}, \ldots\ldots CiC_{xn}\}$ are the total number of component info chunk present in logical component, where x is the number of Layer Info-Chunk that consists of a specific set of Component Info-Chunk and n is the total number of component in the Info-Chunk. A more detailed description is demonstrated in Fig. 11d, where information of the Component Info-Chunk is extracted which correlates with the Layer Info-Chunk.

To understand a community diagram, it is recommended that the nomenclature of Layer Info-Chunk and Component Info-Chunk are correlated. To analyze the information of a behavior, Info-Chunk is filtered by community names. For example, if any changes occurred in the LiC_1, it can influence the engineering objects of LiC_2, LiC_4 and LiC_5 as shown in Fig. 11b. Based on the information obtained from the community diagram, the contribution of a behavior is evaluated during the product modeling. Later, the product model can be updated by initializing behaviors in the content or making changes in the existing behavior of the content.

7.2 Discipline-based Content

In the present manufacturing world, most products are multi-disciplinary. To define an engineering discipline in discipline content, Component Info-Chunk and Layer Info-Chunk are defined similar to behavior content. The flow chart of a discipline content is shown in Fig. 12. Unlike behavior content, discipline content don’t set priority to a discipline. The rest of the steps are similar to the behavior content. The community diagrams are used to extract the relevant information of a discipline by using the Info-Chunk configuration. Later, the product model can be updated by initializing disciplines in the content or making changes in the existing discipline of content.

8 Practical Approach of Conceptual Model

There can be various approaches in taking the conceptual model of information content from concept to reality. In Catia V6, modelica component is used in the logical level of RFLP structure. It is coded using modelica programming language, which is based on the object oriented principles as shown in Fig. 13.

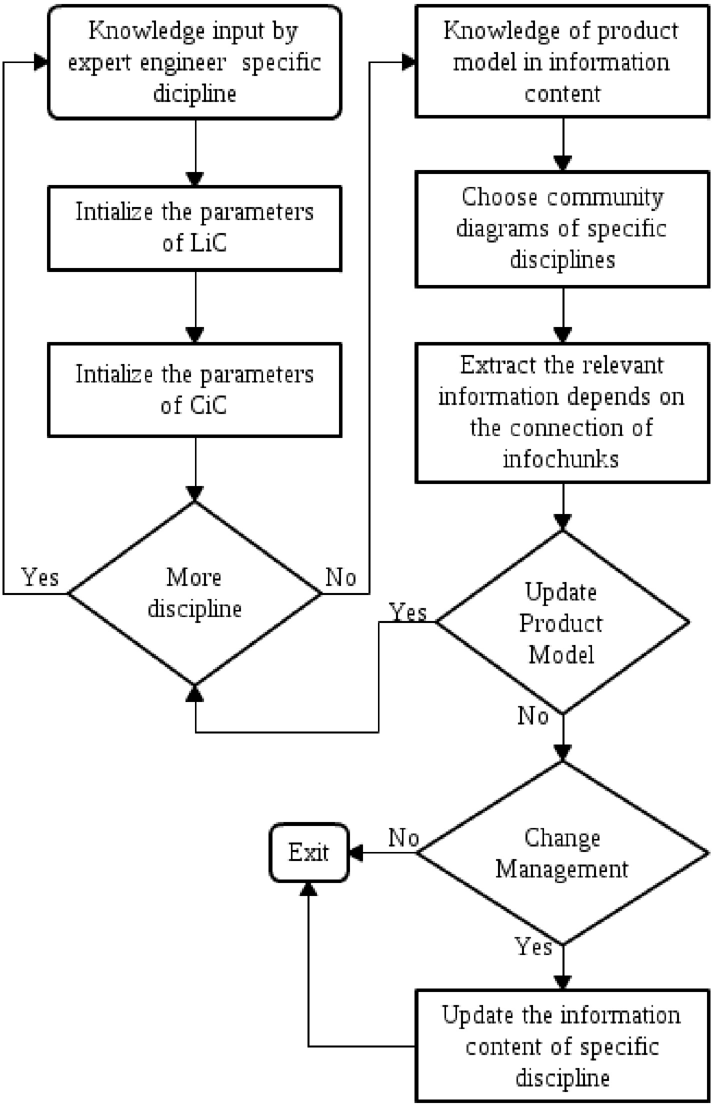


Figure 12
Flow chart of discipline based content

To code an Info-Chunk in the logical level of RFLP structure, it is represented by *InfoChunk* class. It is considered as a base class whose parameters and equations are not yet defined. *ComponentInfo* and *LayerInfo* are the derived class of *InfoChunk* class and is representation of Component Info-Chunk and Layer Info-Chunk respectively.

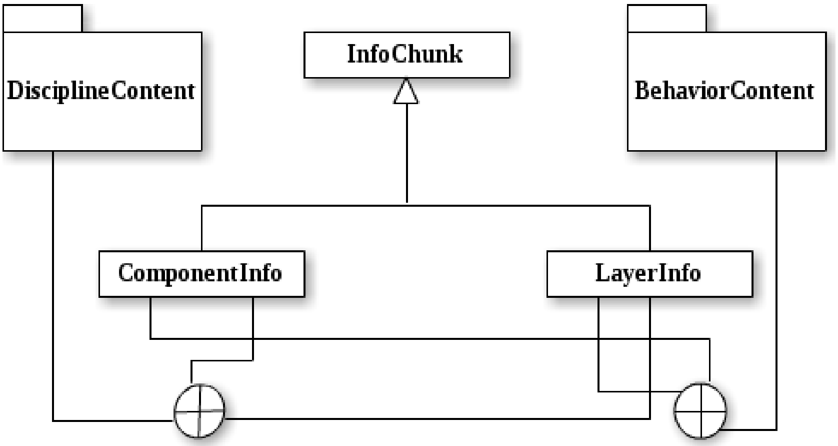


Figure 13
Practical approaches for conceptual models of information content

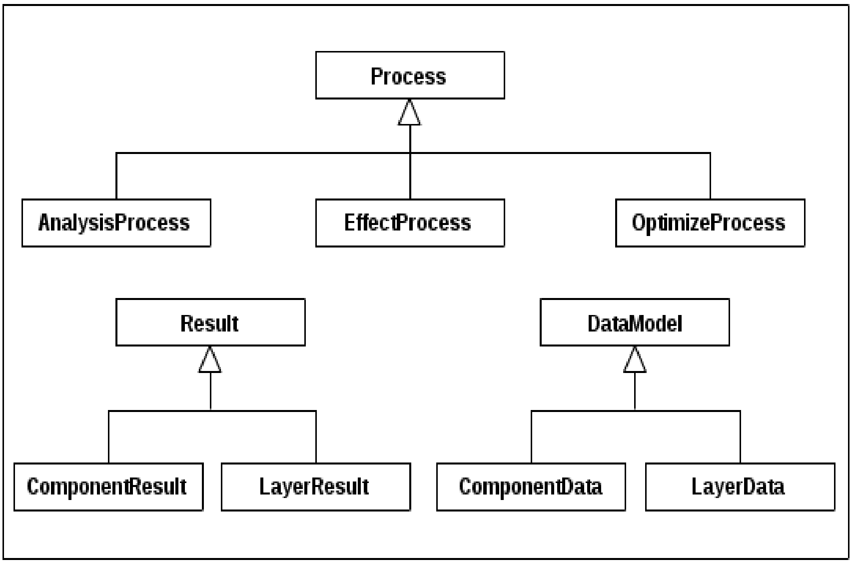


Figure 14
Class diagram for entities in the conceptual model

DisciplineContent and *BehaviorContent* are the package name which can store related *ComponentInfo*, *LayerInfo* and related classes. Furthermore, *ComponentInfo* and *LayerInfo* can initiate the objects of *Process*, *Results* and *DataModel* classes according to the parameters of Info-Chunk. Here, *Process* is the base class and *AnalysisProcess*, *EffectProcess*, *OptimizeProcess* are the derived classes as shown in Fig. 14. Similarly, *Result* is the base class and

ComponentResult, *LayerResult* are the derived classes. *DataModel* is the base class and *ComponentData*, *LayerData* are the derived classes. It is possible to code the Info-Chunk using Simulink component.

Conclusion

This paper proposed conceptual models of information content. The first step is the classification of information content based on the system behavior and engineering discipline. The second step is the introduction of Info-Chunk in the RFLP structure to describe the parameters for the conceptual model of information content. Here, Layer Info-Chunk is placed in the logical level and Component Info-Chunk is placed in the logical component of the RFLP structure. The final step is mapping between Layer Info-Chunk with the Information content to take the correlating decisions. The main purpose of the conceptual model is to store and represent the information of the complex product model into a simplified form so that the engineers can more effectively analyze aspects of the system.

The area of improvements are the parameters of the Info-Chunk, control procedures and integration of information content. Further classification of the information content is also possible. In this paper, the author manually created a conceptual model of information content. The model could be self-adaptive and the Intelligent Virtual space can initialize the parameters of Info-Chunk by algorithms, logic, and rules. In this paper, Info-Chunk is defined for logical and physical level of the RFLP structure. It could be defined for functional level and the next step is to code an API for the conceptual model of information content in the library of modelica.

Acknowledgement

The author gratefully acknowledges his supervisor, Dr. Horváth László, for the guidance while writing this article as well as the financial support of the Doctoral School of Applied Informatics and Applied Mathematics, Óbuda University.

References

[1] L. Horváth, “New Methods on the Way to Intelligent Modeling in Computer Integrated Engineering,” IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society, vol., no., pp. 1359-1364, 7-10 Nov. 2010

[2] L. Horváth and I. J. Rudas, Modeling and Problem Solving Methods for Engineers, ISBN 0-12-602250-X, Elsevier, Academic Press, pp. 46-52, 2004

[3] L. Horváth and I. J. Rudas, “Supporting Lifecycle Management of Product Data by Organized Descriptions and Behavior Definitions of Engineering Objects,” Journal of Advanced Computational Intelligence and Intelligent Informatics, Tokyo, Vol. 11, No. 9, pp. 1-7, 2007

[4] Yatish Bathla, "Structured Organization of Engineering Objects in the Information Content of PLM System" International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisora, Vol. 11, No. , pp. 473-478, 2016

[5] Peter Fritzson, Principles of Object-Oriented Modeling and Simulation with Modelica 3.3: A Cyber-Physical Approach, ISBN 978-1-118-85912-4, Wiley-IEEE Press, pp. 3-25, 2015

[6] Yatish Bathla, "Different Types Of Process Involved In The Information Content Product Model" International Symposium on Intelligent Systems and Informatics (SISY), Subotica, Vol. 14, No. , pp. 99-104, 2016

[7] L. Horváth, I. J. Rudas, "New Approach to Knowledge Intensive Productmodeling in PLM Systems", International Conference on Systems, Man and Cybernetics, ISBN 978-1-4244-0990-7, Vol ,No. , pp. 668-673, 2007

[8] Yatish Bathla, Márta Takács "Evaluating Product System Behavior using SoftComputing in Product Structure Modeling" International Symposium on Applied Machine Intelligence and Informatics (SAMI), Herlany, Vol. 15, No. , pp., 2017

[9] L. Horváth, I. J. Rudas, "Systems Engineering in Product Definition", ISBN 978-1-4799-8221-9, Vol. 13, No. , pp. 181-186, 2015

[10] Frederic Chauvin, Gauthier Fanmuy "Systems Engineering on 3DEXPERIENCE Platform - UAS Use Case", Dassault systèmes

[11] Nancy Lesinski "Systems Engineering in Today´s Automative Landscape", http://blogs.3ds.com, December 7, 2015

[12] Andreas Hofmann, Nils Menager, Issam Belhaj, Lars Mikelsons "Integrated Engineering based on Modelica", International Modelica Conference, Vol, No. 11, pp. 893-901, Sep. 2015

[13] L. Horváth, I. J. Rudas "Towards the Information Content-driven Product Model," International Conference on System of Systems Engineering , vol., no., pp. 1-6, 2-4 June 2008

[14] Dassault Systèmes AB, Dymola Dynamic Modeling Laboratory Getting started with Dymola, pp. 5-25, May 2011

[15] L. Horváth, I. J. Rudas "Product Definition in Virtual Space Using Background information at Description of Engineering Objects," WSEAS TRANSACTIONS on SYSTEMS and CONTROL, Vol. 3, No. 9, pp. 748-758, September 2008