



Requirement Analysis of an OWL Based Disassembly Planning Information Model

Bicheng Zhu* and Utpal Roy

Department of Mechanical Engineering, Syracuse University, New York, USA

Abstract

Disassembly, a process of separating the End of Life (EOL) product into discrete components for re-utilizing their associated residual values, is an important part for the sustainable manufacturing. This work focuses on the modeling of the disassembly planning related information, and develops a Disassembly Information Model (DIM) based on an extensive investigation of various informational aspects of the disassembly planning. In this paper, the requirement analysis for the development of the DIM has been carried out and two categories of the requirements relating to the disassembly domain aspect and the information model aspect are thoroughly discussed.

Introduction

A succinct and comprehensive definition of End of Life (EOL) product is provided by the European Economic Community, which defines the EOL product as “any substance or object which the holder discards or intends or is required to discard” [1]. Normally, the discarded EOL product may or may not be totally obsolete, and a recovery process can be applied to restore the contained value as a form of energy, material or product. Such recovery processes have been more and more studied under the popular paradigm of sustainable manufacturing, which has the objective to carry out economically-sound manufacturing/de-manufacturing processes that maximize the possible profits and minimize negative environmental impacts by utilizing different recovery options, such as recycling, reuse, and remanufacturing.

On the other hand, governments have already started to impose regulatory obligations on manufacturing companies, which mandate manufacturers to set up plans for collection, recycling and recovery for specific types of products. For instance, the Waste Electrical and Electronic Equipment Directive is the European community’s directive on the Waste Electrical and Electronic Equipment (WEEE), which became European law in February, 2003. The Restriction of Hazardous Substances Directive (RoHS) was also adopted in February 2003 by the European Union to restrict the use of certain hazardous substances in the electrical and electronic equipment. In the United States, 25 states have passed legislations, mandating statewide electronic waste (e-waste) recycling and several more states are working on passing new laws or improving the existing laws. All laws, except those in California and Utah use the “Producer Responsibility” approach, where the manufacturers must pay for recycling. Also, 65% of the U.S. population has been covered by a certain state level

e-waste recycling law since 2003 [2].

Both the potential economic profits and the regulatory laws motivate the study of the EOL product recovery modeling and implementation. Almost for all the different recovery strategies, some level of disassembly process is involved. In this sense, carrying out the disassembly process “optimally” plays a critical role in the entire process of the EOL product recovery. Over the years, various methods ranging from network theory to mathematical programming have been applied in the domain of product disassembly [3]. Unfortunately, very few researches look into the problem from the information aspect, which is the bottleneck of the current disassembly related research. Disassembly planners have limited knowledge on what information is critical in the planning of the disassembly process, how to access this information, and, finally how to utilize the updated on-site information (which is unknown in the beginning of the disassembly process) for dynamically adapting the “optimal” disassembly process plan. Also, every EOL product is unique and has to be treated individually, which further aggravates the above mentioned problems.

Fortunately, with the advent of the internet, come computer aided tools are available to overcome the obstacles mentioned above. Two specific technologies, Internet of

***Corresponding author:** Bicheng Zhu, Department of Mechanical Engineering, Syracuse University, New York, USA

Accepted: June 18, 2020

Published online: June 20, 2020

Citation: Zhu B, Roy U (2020) Requirement Analysis of an OWL Based Disassembly Planning Information Model. J Robotics Autom 4(1):131-142

Things (IoT) and Life Cycle Units (LCU), have already been discussed in the disassembly research community for ideas like future cloud-based remanufacturing [4] and semantic recovery information service [5]. Briefly, IoT provides a network to connect different physical objects, which allows them to be sensed and controlled remotely across existing network infrastructure, creating opportunities for more direct integrations of the physical world into computer-based systems, and resulting in improved efficiency, accuracy and economic benefit. LCU, on the other hand, is developed specifically for the product disassembly process. As mentioned before, in a disassembly factory, different products arrive continuously for disassembly, and individual decisions regarding optimal disassembly sequences have to be made for every product. It is difficult to predict any pre-defined disassembly process sequences a priori, so the detailed information on how to disassemble each arriving product is needed. LCU is proposed for decentralizing that information by integrating a physical device named Life Cycle Units (LCU) into every product. The LCU stores information needed for disassembly. Once enough disassembly information about a product is present, the optimal disassembly sequence can be generated based on the actual physical status of the EOL product. Combining the LCU and IoT technologies together, individualized EOL product information could be sensed and collected by LCU and transferred to the central Product Lifecycle Management (PLM) system through the IoT network. This could be used to tackle the problem of disassembly information bottleneck.

In this reported work, we develop a Disassembly Information Model (DIM) that can be integrated into the future sustainable and smart manufacturing environment, for efficient disassembly planning activities. As the first part of this work, detailed requirements regarding the DIM are identified and analyzed in this paper, which looks into both of the disassembly domain aspect and information modelling aspect. An overview of the developed DIM is also briefly presented. A layered Information Model (IM) development methodology is proposed to address the reusability-usability trade-off problem. The developed DIM is further implemented into the Web Ontology Language (OWL), through which relevant information can be computationally analyzed and utilized.

Literature Review

A comprehensive review on the scientific background and an establishment of the technical terminologies related to this work are carried out in this section. Sequentially, two major topics, the EOL Product disassembly planning problem and the Information Model & Ontology, are discussed in detail. The findings and observations from the literature review are also discussed, which bring to light the potential opportunities for disassembly planning research.

EOL product disassembly planning

Different planning approaches and methods have been proposed for the disassembly processes; they can be broadly divided into the following categories:

Graph-based approach: Graphs usually represent the structure of a system, process, product, organization, etc.

They can be considered as an abstraction of the reality. Graph theory has been used as a powerful tool to solve the problems of disassembly planning, and representation models like connection diagram and AND/OR graph are usually utilized in such methods. The characteristics and functions of a disassembly system are explicitly expressed in the graph, and different searching algorithms are applied to find all the feasible disassembly sequences according to the topological, geometrical and technical constraints. Different strategies are further applied to locate the optimal sequence with consideration of the plan effectiveness and cost-effectiveness. Several outstanding graph-based approaches are briefly discussed below.

Penev, et al. [6] used AND/OR graph theory and methods of dynamic programming for the generation and evaluation of the feasible disassembly plans. A new economic model is introduced to determine the optimal disassembly depth. Zhang, et al. [7] developed a graph based heuristic approach for the generation of disassembly sequences from CAD system directly. They proposed a Component-Fastener Graph to analyze the product assembly relationship and a searching for cut-vertex and decomposition of the EOL product into several subassemblies is further applied on the graph to simplify the disassembly analyzing process. Murayma, et al. [8] described the disassembly sequence generation using the idea of information entropy and heuristics to replace components at maintenance stages. The advantage of this method is primarily in the reduction of searching time and searching places for disassembly sequences. The author also developed a software tool integrated with a CAD system and carried out an experiment for an electric drill using the tool. A graph-based information modelling system to represent the process for disassembly and recycle planning of consumer products was proposed by Kanai, et al. [9]. Four kinds of graph have been presented: (1) A configuration graph of sub-assemblies or fragments; (2) A connection graph between parts and materials; (3) A process graph of disassembly, shredding, and sorting activities; (4) A retrieval condition graph. Rules and procedures for transforming the models of these activities are uniformly formulated. A vacuum cleaner is used as an example to demonstrate the proposed graph-based method. Lambert [10] proposed a Linear Programming (LP) model, based again on the AND/OR graph, to the disassembly planning problems. The LP model tries to find the optimal disassembly sequence based on maximizing the total value of the retrieved parts/subassembly and minimizing the total disassembly operation cost associated with them.

Petri net-based approach: Besides the traditional graph-based disassembly analysis approach, Petri-Net (PN), as a graphical and mathematical tool, provides a uniform environment for modelling and analyzing both static and dynamic discrete events. They provide a very promising method for disassembly sequence generation.

Zussman, et al. [11] proposed a complete and mathematically sound Disassembly Petri Net (DPN) approach to model the disassembly processes. In their work, the detailed construction and advantages of the proposed DPN have been discussed, and a DPN based searching algorithm has been

proposed for the generation of the disassembly plan. They further extended this work [12] and proposed a design and implementation system for an adaptive process planner for disassembly processes. The system also incorporates the uncertainty issue caused by the different product conditions.

Moore, et al. [13] developed an algorithm for automatically generating a DPN from a disassembly precedence matrix. The DPN representing the specific precedence relationships among parts can be derived from a CAD representation of the product. A Reduced Reachability Tree algorithm has been further proposed to identify the near-optimal disassembly process plan from using the DPN.

AI based approach: Many attempts have been made using AI techniques (Genetic algorithms, ant colony methods, fuzzy logic, neural networks, etc.) in the disassembly sequence optimization. The objective is to reduce computing time by searching the best disassembly sequences without analyzing all the possible alternatives. Several examples are discussed as below:

An example of the use of fuzzy logic in disassembly planning is proposed by Chevron, et al. [14]. The main goal is to find the disassembly sequence requiring the minimum completion time while taking into account the fuzzy model of the processes and the constraints such as available resources, destruction modes, etc. The problem of the generation of disassembly sequences is approached as a travelling salesman problem (the traveler is the product and the cities are the operations with their processing times). A modified branch-and-bound method is used with an objective function evaluated according to fuzzy parameters.

Hsin-Hao, et al. [15] proposed a Neural Networks approach to the planning of disassembly problem. The generation of sequences is again viewed as a variant of the travelling salesman problem: To find the sequence of components to be disassembled (cities) having the greatest profit (the shortest distance). This problem is approached using a Hopfield Neural Network. As input, an N by N matrix of neurons is used: The rows of the matrix indicate the disassembly operations to be scheduled, and the columns the disassembly sequences.

Lambert [10] proposed a Linear Programming (LP) model to the disassembly planning problems. The LP model tries to find the optimal disassembly sequence based on maximizing the total value of the retrieved parts/subassembly and minimizing the total disassembly operation cost associated with them.

The above disassembly-planning model formed the basis of the disassembly sequencing related research. Some recent works have been extending on various aspects of the theory for certain specific issues (e.g. Yu, Ilgin, M.A, and Gao Y) [16-18], but the core concepts remain the same. Also, several important observations can be identified here:

1. Although different researchers proposed different representation models, the involved information (product, process, etc.) shared similarities among different methods. The reuse of these concepts has not been explored, which could have made the develop-

ment processes of the Computer Aided Disassembly Planning applications less time consuming.

2. The representation model is not appropriate for the full support of the lifecycle information collection and analytics, under the paradigm of smart product and IoT. Specifically, we need an information mechanism, which can treat each individual EOL product as unique and public information source and connect it to relevant other information, necessary for the disassembly planning application.

Information model and ontology

The Information Model, sometimes called ontology, is the consensual modelling of concepts and relationship in a domain of interest. Over the years, researchers have contributed to the development of IM or ontology in the domain of manufacturing, with different focus. Some notable works are reviewed below.

Leimagnan, et al. [19] developed the Manufacturing Semantic Ontology (MASON) to formally capture the concepts related to the manufacturing industry. The semantics related to entity, resources and operation were captured in formal logic using web ontology language (OWL). Two applications about automatic cost estimation and the semantic-aware multi-agent system for manufacturing were discussed to demonstrate the usefulness of the proposed MASON ontology.

Xiaomeng [20] selected the field of Design for Manufacturing (DFM) and three primary aspects are investigated. First, a generalized DFM ontology is proposed and developed, which fulfills the mathematical and logical constraints needed in the domain of DFM. Second, the means to guide users to the proper information and integrate heterogeneous data resources is investigated. Third, a decision support tool is developed to help designers consider the design problem in a systematic way based on the developed DFM ontology.

Pavan [21] developed an ontology called the Design Activity Ontology (DAO) to explicitly represent the design activity that can cover phases of the design process from conceptual phase through detail design phase. The ontology provides a formalized and structured vocabulary for design activities and it enables design processes to be modeled, analyzed and optimized in a consistent way.

Kim, et al. [22] proposed a collaborative assembly design framework that offers a shared conceptualization of assembly modeling and an Assembly Design Ontology (AsD) is developed to capture the joining intents of a product. AsD is claimed to serve as a formal, explicit specification of assembly design so that it makes the assembly knowledge both machine-interpretable and sharable.

Some industrial efforts have also been reported to the development of the manufacturing related Information Model, a notable development in this field is led by NIST. One of their work is the NIST's Core Product Model (CPM), which a unified modeling language (UML) based model intended to capture the full range of engineering information commonly shared

in product development [23]. CPM focuses on modeling the general product information and excludes the information which is domain specific. NIST further developed another information model called “Open Assembly Model” (OAM) [24] which extends CPM. Along with the structural information, it represents the function, form, and behavior of the assembly, and defines a system level conceptual model.

Recently, NIST also proposed a disassembly information model [25] and this is the first attempt to develop disassembly related information model. However, three issues have not been well addressed in the NIST disassembly information model: (1) The focus of the NIST disassembly information model is on the reuse, maintenance, and recycling aspect and the information regarding to the disassembly planning is not sufficiently modeled; (2) The NIST disassembly information model remains in the conceptual stage and the model implementation under the paradigm of smart product and IoT has not been carried out; (3) The handling of reusability/usability tradeoff issue (refer to section 4.2 for detailed descriptions) and the uncertainty issue is not discussed.

Disassembly Planning Information Model Requirement Analysis

The requirements for the Disassembly Information Model development consist of two parts. The first part relates to the domain aspect, which indicates that the developed DIM should contain relevant informational elements necessary for the disassembly planning process. The second part relates to the modeling aspect, which indicates that the modeling process is not just information collection; rather the information needs to be well modeled in a way that it can be useful as well as reusable.

Modeling of Relevant Informational Element (R1)

The information required for the EOL product disassembly planning can be broken down into four categories: Product related, process related, uncertainty related and component degradation related (Figure 1). The informal description of each category is presented below:

Product related information: The product related information describes the characteristics of the EOL product which

needs to be disassembled. Relevant concepts or terminologies in this aspect include product, component and liaison, and they will be informally described below.

Product and component: In general, a product is an artifact or substance that is manufactured for sale. In any disassembly process, the product represents the input to the disassembly process and it may consist of a number of discrete parts, which are called components. A component is a material entity that can be separated from a product through disassembly processes, without altering the component’s intrinsic property (like mass, density, etc.). Furthermore, a component cannot be further separated via non-destructive disassembly processes.

In the domain of disassembly planning, a further specification of the component according to their characteristics is critical. From a high level, components can be classified according to different aspects like material composition (homogeneous or composite), functional type (connecting function or non-connecting function) and component complexity (atomic component or complex component). In detail, the following types of component are highlighted:

Homogeneous component: Is a component consists of only homogeneous materials. Housing and cover of a blender are the typical examples of the homogeneous component.

Composite component: Is a component consists of different non-homogeneous materials linked in an irreversible way, such as a sandwich structure. The laminated glass, which is constructed by combining two panes of glass fused together with a middle layer of Polyvinyl Butylenes Film (PVB) acting as a bonding agent, is an example of composite component.

Connecting component: Is a component whose primary function is to connect other components. Different fasteners fall under this category.

Complex component: A complex component is a cluster that consists of a set of components, which cannot be separated from the whole without damaging certain component permanently. Examples of such component can be printed circuit boards and electrical cables.

Liaison: Components are physically linked by liaison, which restricts the freedom of motion of the components in-

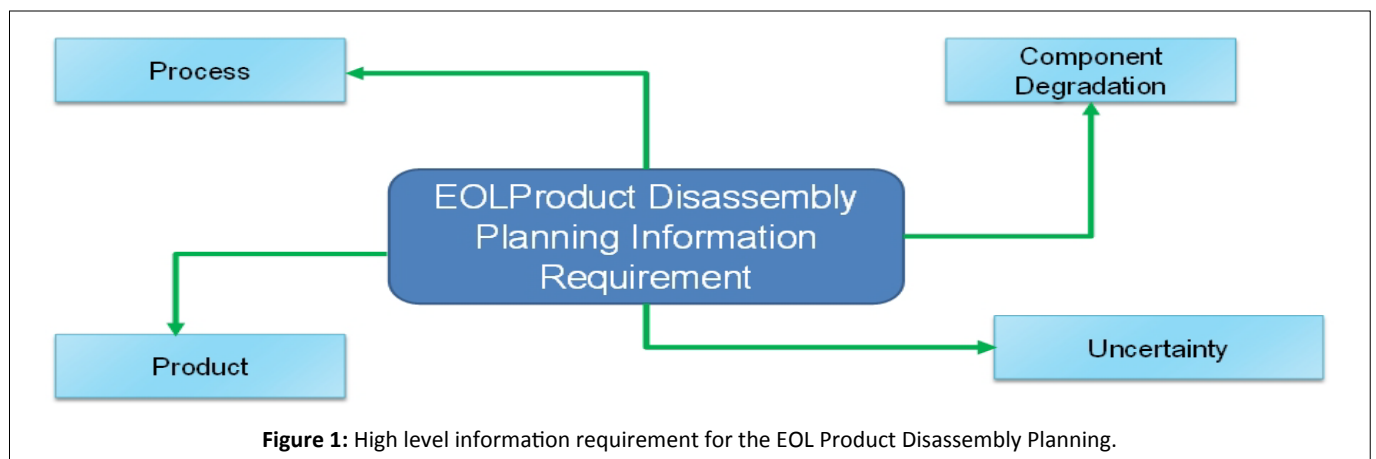


Figure 1: High level information requirement for the EOL Product Disassembly Planning.

volved. The liaison concept can be classified into two main types to reflect the different properties of the liaison. The main liaison types are:

Component contact: Such liaison represents the relationship between components where the involved components are connected with each other without any application of external forces. We call this type of connection “component contact” and it is formed through connections between component’s geometric entities like a vertex, an edge or a surface. Examples of such case could be a cube resting on a panel (surface contact).

Component connection: Such liaison represents the relationship between components where a connection is established through a certain connecting component. Example of such case could be a blender housing connected with a base panel by a set of screws (connecting component).

From the discussion above, two types of Information Modeling requirements related to the product domain aspect have been identified. The first one is the modeling of product hierarchy: A product is composed of different components which are hierarchically organized by aggregating them into subassemblies. Thus, the part-whole relationship needs to be modeled in the product domain Information Model. Second modeling requirement relates to the topological arrangements of components (or say product structure), which are realized by different component liaisons. Information related to how components are connected to each other for achieving the final product should be provided. If the liaison belongs to the type of component connection, at least three entities are then involved: Two components are connected through one connecting component. Thus, modeling of n-ary relationship ($n > 2$), which involves more than two entities, should be supported.

Process related information

The disassembly process accomplishes the basic transformations of the product’s physical states and it can be divided into three different levels as follows:

Task level: Task represents the most abstract type of disassembly process, which only specifies the target component to be disassembled. An example of a disassembly task could be “detaching blender housing component” or “disassembling the screw from the PC motherboard”. A sequential aggregation of disassembly tasks will provide a high level disassembly plan.

Operation level: An operation represents the detailed process steps necessary to achieve a certain task. The operation may not only include disconnection process, they may also include the movement operations necessary to transfer the subassemblies to a different location and other supplementary operations such as cleaning, fixturing, tool exchanging, product reorienting (to guarantee access or stability), and testing.

Action level: An action represents the specific atomic process steps required to achieve a certain operation. An important characteristic of an action is that it is performed without

the goal to directly change the object state [26]. It means that an action alone should not be sufficient to change any part attributes or disestablish of liaisons. As an example, a “movement” operation involves possibly two actions: “Motion” action and “grip” action. However, neither the “motion” action, nor the “grip” action alone changes the state of the object (A “motion” action will not make any difference on the component unless it is combined with a “grip” action).

Along with the detailed process classification, another important process related requirement is the ability to represent all the feasible disassembly process sequences explicitly. In other words, this requires the development of an Information Model that can mimic the traditional graph based disassembly process representation, such as the state change graph. This requirement utilizes again n-ary relationship because a state change normally involves three objects: Pre-state, goal-state and process step.

Uncertainty related information

Unlike the assembly process, the disassembly process has various uncertainty issues that need to be considered. Thus, extra information is needed for such uncertainty handling. Two types of uncertainty are considered in this paper: (1) Component/assembly function uncertainty and (2) Operation uncertainty.

Component/assembly function uncertainty: Each component or assembly might associate with a primary function, which contributes to the product overall function. Such function may not be available when the EOL product becomes obsolete. Such functioning/non-functioning information is critical in the disassembly planning process, and can only be revealed gradually during the disassembly process.

Operation uncertainty: During the disassembly process, certain operation such as unscrewing might not be successful due to the bad component conditions such as deformation or corrosion. In such cases, extra special operations are necessary to handle the situation, which will incur a higher cost. Since this information is also unknown at the beginning of the disassembly process, it is considered as an operation uncertainty.

Both cases will be handled using the Bayesian Network (BN), which consists of a Directed Acyclic Graph (DAG) and a set of local statistical distributions [27]. Two important information elements are necessary for the Bayesian theorem based uncertainty handling: (1) The component/assembly influence dependency and (2) The Conditional Probability Table.

Component/assembly Influence dependency: Is a directed acyclic graph (DAG) representing the function dependency among components/assemblies. Figure 2 is a simple DAG example describing that the function of the “Fan Assembly” is conditionally dependent on the function of the “Motor” and function of the “Rotor Shaft”. Such information demonstrates the propagations of a function failure in the EOL product, and is critical in the adaptive disassembly planning.

Conditional Probability Table (CPT): Consists of a set of

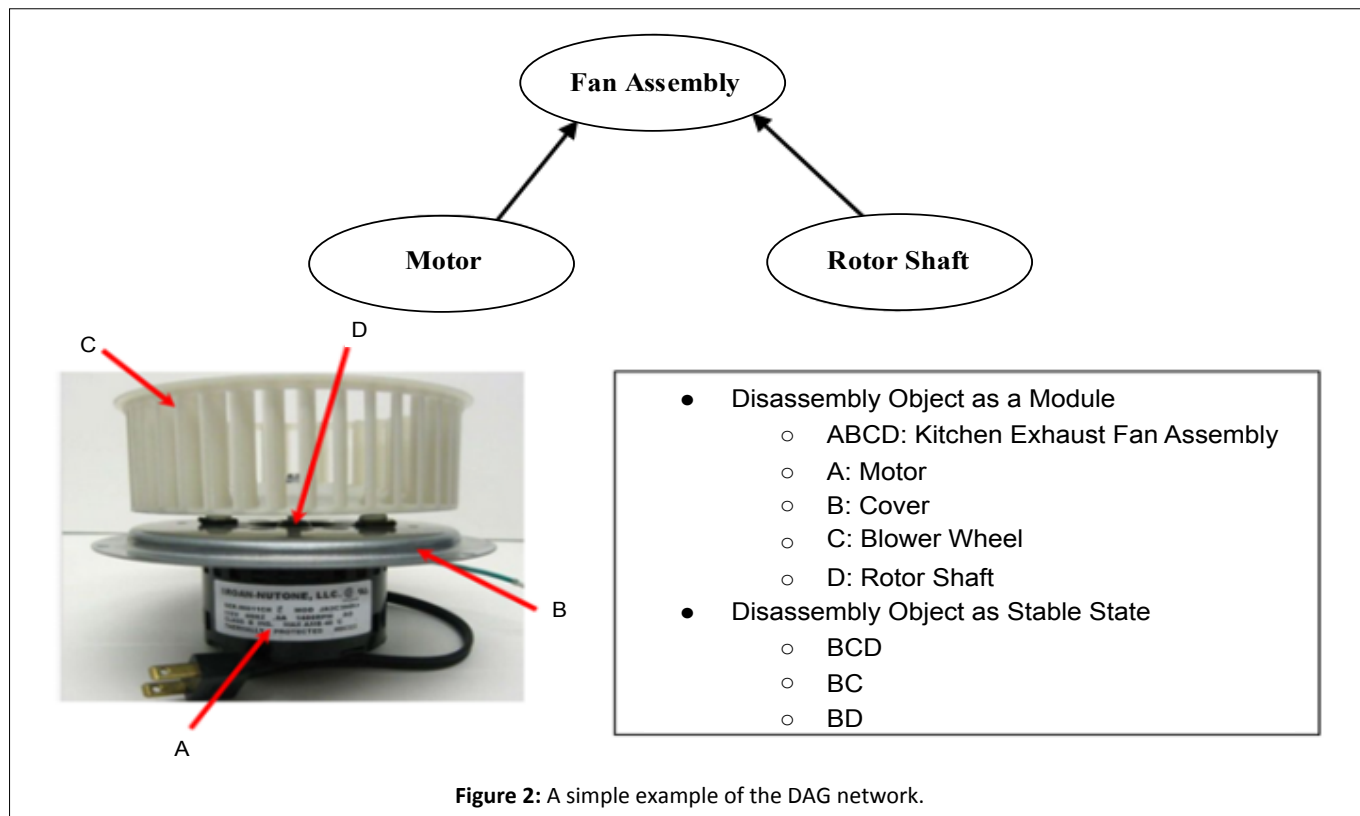


Figure 2: A simple example of the DAG network.

Table 1: A simple CPT example.

	Motor	Function		Not Function	
	Rotor Shaft	Function	Not Function	Function	Not Function
Fan Assembly	Function	0.8	0	0	0
	Not Function	0.2	1	1	1

discrete (not independent) random variables to demonstrate the marginal probability of a single variable with respect to the others. A simple CPT applied to the example in Figure 2 is shown in Table 1. It implies that the probability of the “Fan Assembly” to be functional is conditionally dependent on two other variables: the probability of the “Motor” to be functional and the probability of the “Rotor Shaft” to be functional. As an example, when the motor is functional and the rotor shaft is not functional, the probability of the fan assembly to be functional is 0.

Information regarding the degradation aspect

Component/assembly degradation is also a critical issue in the planning of disassembly. Degradation is a gradual change in properties (like tensile strength, color, shape, etc.) of the component, which usually does not affect the overall function of a component until it reaches a critical point. However, degradation does affect the economic quantification of EOL product or its component. For example, although some subassembly might work fine (functional) after the function testing, the associated reuse value still could be lower than the expected average reuse value (the subassembly is close to failure) or higher than the expected average reuse value (the subassembly still has a long remaining useful life time). Information regarding such remaining useful life estimation

should be available for an effective disassembly planning process.

The remaining useful life time estimation is a challenging research problem and in this paper we use the fuzzy logic based approach to quantify the component/subassembly reuse value through a set of fuzzy linguistic variables and a set of heuristic rules. The detail of reuse value estimation using fuzzy logic will be presented in detail in the following chapter (chapter 5). Here we only summarize the important information elements necessary for carrying out the fuzzy logic based reuse value estimation.

Age: Age represents the service time of a component or a product. This information could be different among different components in the one product (component replacement during the maintenance). Usually, high age indicates a lower reuse value.

Condition parameter: Age is an indicator variable for estimating the remaining component useful lifetime. However, it is assumed that the component or subassembly is servicing under certain controlled operational conditions. If the user is abusively using a certain product or a product is operating under severe external environments, the age of the component/product alone can no longer properly indicate the remaining useful life time. Certain condition parameters (like operation

Table 2: A summary of the disassembly domain information requirements.

Product Aspect Information Requirement	
Basic Terminology	Product
	Component <ul style="list-style-type: none"> ● Homogeneous component ● Composite component ● Connecting component ● Complex component
	Liaison <ul style="list-style-type: none"> ● Component Contact ● Component Connection
Modeling Requirement	Product Hierarchy <ul style="list-style-type: none"> ● Part-whole relationship
	Product Topology (Component Liaison) <ul style="list-style-type: none"> ● N-ary relationship
Process Aspect Information Requirement	
Basic Terminology	Process <ul style="list-style-type: none"> ● Task ● Operation ● Action
	Modeling Requirement
	Feasible Process Sequences <ul style="list-style-type: none"> ● N-ary relationship Process Hierarchy <ul style="list-style-type: none"> ● Part-whole relationship
Uncertainty Aspect Information Requirement	
Modeling Requirement	Bayesian Networks <ul style="list-style-type: none"> ● Conditional Probability Table ● Component/assembly Influence dependency
Degradation Aspect Information Requirement	
Modeling Requirement	Fuzzy Logic <ul style="list-style-type: none"> ● Linguistic variable <ul style="list-style-type: none"> ○ Age, Condition Variable, Market Demand ○ Membership function ○ Fuzzy term ● Fuzzy Rules

noise, corrosion, etc.) should be included for the estimation.

Market demand: Reuse value is also dependent on the market demand. A higher demand normally will increase the average reuse value and a lower demand will decrease the reuse value despite of the conditions of the product/component.

The above input variables will be modeled as linguistic variables, which is suitable for fuzzy reasoning. Extra informational elements related to the linguistic variable are thus necessary, which include membership function and fuzzy term definition. On top of that, the support of heuristic based fuzzy rules should be provided in the developed DIM for the fuzzy reasoning process.

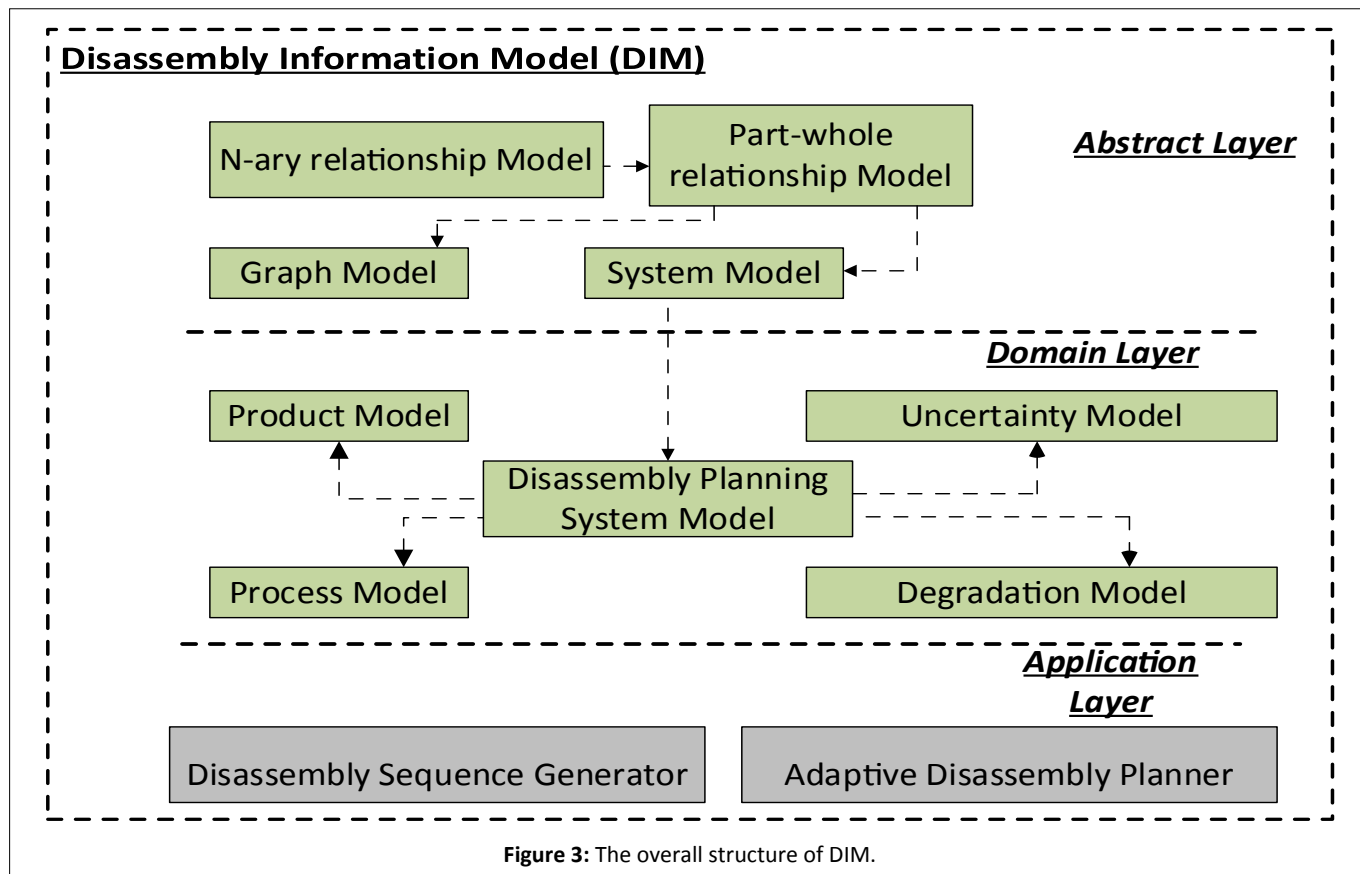
In summary, disassembly planning related information requirements can be summarized as shown in Table 2 below.

Modeling Aspect Requirement (R2)

Generally, an Information Model should meet two major goals: (i) To be usable and (ii) To be reusable. IEEE Standard defines reusability as “the degree to which a software module

or other work product can be used in more than one computing program or software system” [28]. Similarly, Information Model reusability can be defined as “the adaptation capability of an Information Model to arbitrary application contexts”, including those contexts “that were not envisioned at the time of the creation of the Information Model” [29]. It should be understood that it is not feasible and desirable to develop an IM that is equally fitting to all application contexts [30]; rather the goal of reusability is to design an IM which can be extended and adapted to a large number of applications in the domain of interest.

On the other hand, usability in software engineering denotes the degree to which the software component is useful for a specific task or application. When applying the concept in the context of Information Modeling, one important difference should be noted: An IM is rarely ready for use and always will be adapted and refined to a knowledge base for the tentative application. Therefore, the goal of IM usability can be rephrased as minimizing “the effort required to customize the IM so that it can be used by humans or machines in a given application context” [29].



It should be noted that IM reusability and usability requirements are contradicting each; increasing the reusability of knowledge implies the maximization of the use of this knowledge among several kinds of tasks and the resulting IM would be general in nature; increasing usability implies providing all information related to a specific task and the resulting IM would have redundant information for other tasks and thus would not be appropriate. Consequently, it is difficult to simultaneously achieve high degrees of usability and reusability: Specializing in one kind of task makes the IM more useable for this particular task, but it also decreases the likelihood of its reusability; a highly abstract IM, on the other hand, may be applicable to a variety of different tasks, but it is unlikely to be proved useful for any of these tasks without extensive modification and detailing. This is known as the reusability-usability trade-off problem in the literature [31]. In this research, a layered IM development methodology is adopted to address this issue, and it is present in detail in the next section.

Overview of the Disassembly Planning Information Model

Following the discussion in previous sections, it is evident that DIM should comprise of the information related to the product, process, uncertainty and degradation aspects and the modelling of them involves certain information modeling patterns like n-ary relationship, part-whole relationship, etc. Also, DIM should achieve a balance between IM usability and reusability. Thus, a layered modelling methodology has been utilized, in which DIM has been subdivided into three

important layers (Figure 3), with an intention of separating general knowledge into different level of abstractions. Also, a “minimal ontological commitment” [32] guideline is followed, which means each layer holds only concepts/relationships and axioms that are essential for the function of the current layer. Information that is not essential for the layer’s purpose is sourced out to lower layers. Details of each layer are presented as follows:

Abstract layer

The Information Models in the abstract layer hold the fundamental modeling concepts, which are independent of a particular problem or domain and can therefore be universally applied. They describe the design guidelines (design pattern) for the construction of the other sub models in the DIM. Models like n-ary relationship, part-whole relationship, graph model and system model belong to this layer.

Domain layer

The Information Models in the domain layer capture the knowledge related to a domain of expertise, such as the disassembly planning in our case, and they generally don’t target on solving a specific problem or task, but rather provide a domain knowledge foundation for a range of different applications. Thus, the Information Model residing on this layer is more specific than those in the abstract layer, but less specific than those in the lower layer (application layer). The majority of the required disassembly domain information discussed in section 3-1 (product, process, etc.) are implemented in the models in this layer.

Application layer

Represents the most specific Information Model which is directly usable for a certain disassembly planning application. We developed two disassembly planning applications: (1) Disassembly Sequence Generator and (2) Adaptive Disassembly Planning in this work to show how to extend the abstract and domain layer information for the development of disassembly planning related applications (details not included in this paper).

Such a layered DIM development methodology takes the IM reusability-usability trade-off problem into account. The abstract or general knowledge is modeled in the sub models located on the top layer of the DIM. They provide various design patterns which can be reused in various application contexts and normally are not directly usable due to the high abstraction. On the other hand, knowledge in the models residing on the lower layer is ready to be used, but is usually application specific and thus hardly to be transferred to other applications. Information Models in each layer of the DIM contain knowledge with certain degrees of reusability and us-

ability and the usability of the knowledge normally increases with descending reusability when navigating from the top to the bottom layers of DIM.

Another important modeling issue lies in the fact that systems are often too complex to be understood and handled as a whole. If we take an EOL product disassembly system as an example, related information could spread over several aspects like product, process, uncertainty, etc. We thus use a technique for complexity reduction that is widely used in the field of system engineering called the adaptation of viewpoint [33]. A viewpoint is an abstraction of the whole system restricted to a particular set of concerns. Adopting a viewpoint makes certain aspects of the system 'visible' while making other aspects 'invisible'. This way, we can focus on the specific viewpoints of a system, which is of special interest and address separately to the issues in other system viewpoints.

We thus introduce the concept of **Aspect System** in the system model in the abstract layer, which plays a fundamental role in the modeling of the complex disassembly planning system. A general idea is presented in Figure 4 below: The

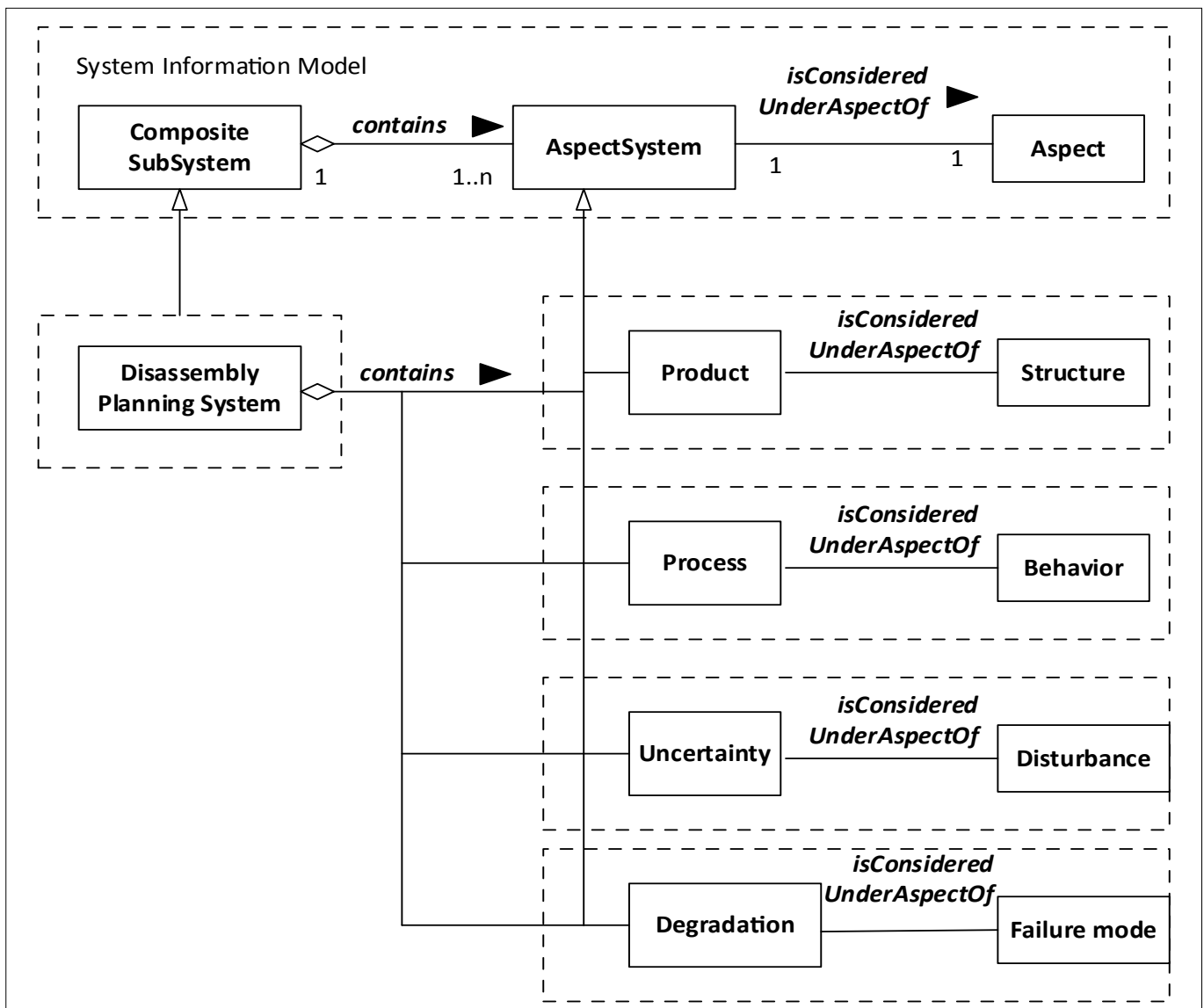


Figure 4: Aspect system utilization in the modelling of the Disassembly Planning System.

Disassembly Planning System class is being modeled as a specialization of the **Composite SubSystem** class, which contains several **Aspect Systems (Product, Process, Uncertainty and Degradation)**. Each of the **Aspect System** models the **Disassembly Planning System** under specific viewpoint (**Aspect**) and is a standalone sub model. As an example, the **Product** class is a subclass of **Aspect System**, which targets on modeling the structural aspect of the disassembly system. Similarly, the **Process** subclass focuses on how to carry out each disassembly steps in order to achieve a certain component detachment task, thus it describes the behavior aspect of the overall disassembly system. The whole **Disassembly Planning System** is an aggregation of the four **Aspect Systems**. The advantage of this design pattern is that the aspect systems can be used and maintained independently of the overall system.

The detailed information regarding each of the sub models is presented in the second part of this work (An OWL based Disassembly Planning Information Model for Sustainable Manufacturing, Part 2: Modeling and Validation). The detailed information can also be found at [34].

DIM Web Ontology Language Implementation

DIM represents a conceptual information model in the domain of disassembly planning and it has been implemented into the Web Ontology Language (OWL) for formal machine reasoning and interpretation. Before presenting the DIM OWL implementation, the background and reason to use OWL as implementation language is briefly discussed below.

Why using OWL for DIM implementation

Under the paradigm of IoT and smart manufacturing, a new concept called smart product has been introduced which has the ability to (i) Sensing the environment, (ii) Store the data in the digital memory, (iii) Communicate with other facilities or smart products and (iv) Carry out knowledge reasoning. This gives new opportunities for solving the problem of EOL product disassembly in which product life cycle data plays a fundamental role. In this work, we focus on the information aspect and two important related requirements (with OWL solutions) are listed below:

R1: The product embedded systems like LCU usually require a fast processing with restricted resources and thus the implementation syntax needs to be compact.

Solution: OWL is an XML based textual file, which can be processed or reasoned by lightweight existing query and reasoning plugins.

R2: The implemented DIM is going to be published as a formal Disassembly Information Model on the Web with the possibility to be connected to the other domain applications. Thus, the implementing language should provide easy mechanisms for connecting DIM with other “things” on the Internet without pre-assumptions.

Solution: In OWL, each objects defined in the DIM is annotated with a Unified Resources Identifier (URI), which helps other applications to access the information and connect to DIM through the web. On top of that, different from the traditional database approach, OWL adopts the open-world assumption (the truth value of a statement may be true irrespective of whether or not it is known to be true), which facilitates the further extensions of the proposed.

DIM: users can add knowledge to DIM as long as it did not semantically contradict with the current definition and such validating process can usually be carried out automatically by certain reasoners like Pellet or Hermit.

R3: The implemented DIM should facilitate the automated knowledge reasoning process.

Solution: Semantic Web Rule Language (SWRL) is supported by OWL, which extends the set of OWL axioms to include Horn-like rules. It thus enables Horn-like rules to be combined with an OWL knowledge base to provide flexible and powerful knowledge reasoning capabilities.

Implementation of DIM in OWL

Protégé 4.0 [35], as an OWL editor, is used in this work to develop formal DIM OWL implementation. The complete OWL implementation code can be accessed from: <http://disassembly-planning-ontology.sourceforge.net>.

The DIM is a set of models distributed in three hierarchical layers and certain dependence or aggregation relationship exists among different submodels. As an example, the “Disassembly Planning System” submodel, being the most complex submodel in the domain layer, is actually an aggregation of four domain submodels (Product Model, Process Model, Uncertainty Model and Degradation Model). On top of that, the “Disassembly Planning System” model itself is a system in nature, thus it is dependent on the System Model from the abstract layer.

DIM is thus implemented into 11 OWL sub models shown in Table 3. Each OWL file represents a submodel in the DIM and is classified into different layers (abstract layer, domain layer or application layer). The dependence or aggregation relationship exists among models is realized by importing the

Table 3: DIM OWL Implementation.

Abstract Layer	Domain Layer	Application Layer
N-ary-relationship.owl	DisassemblyPlanningSystem.owl	DisassemblySequenceGenerator.owl
Part-whole.owl	Product.owl	AdaptiveDisassemblyPlanner.owl
Graph.owl	Process.owl	
System.owl	Uncertainty.owl	
	Degradation.owl	

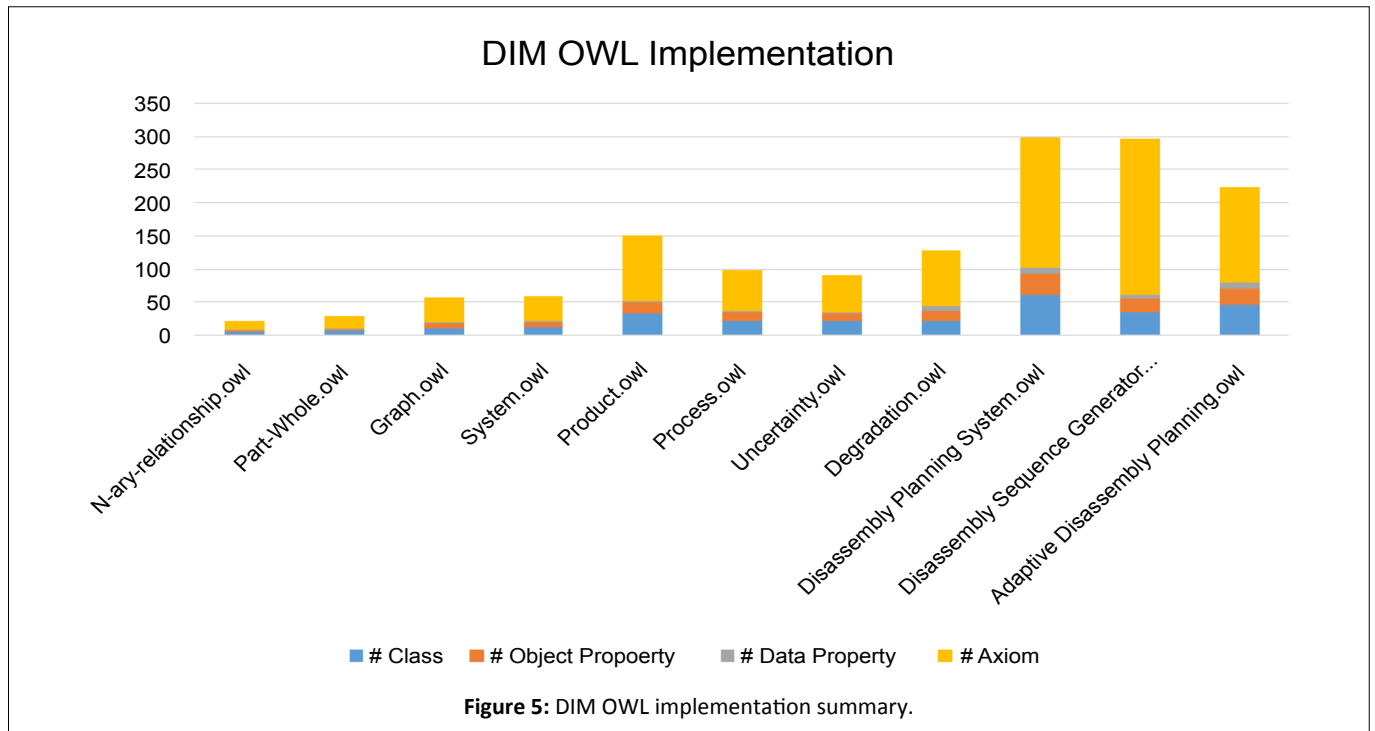


Figure 5: DIM OWL implementation summary.

related sub models into the existing model. Figure 5 provides an DIM OWL implementation summary regarding the certain schema metrics, like number of classes, number of object properties, number of data properties and axioms.

Analyzing and validating the quality of an information model is always challenging task and enormous recommendations have been suggested in the literatures. However, there are few consensus being established as a standard. Thus, it is difficult to quantify the degree of quality due to the absence of generally accepted key measures assessing an agreed set of quality indicators. On the other hand, more and more researchers believe that the development of Information Model should follow the “Kaizen” approach, which suggests “continuous improvement during its lifecycle, both in the development stage and the utilization stage”. In other word, Information Model is validated through continuously being used in different applications, and through continuously updating and modifying.

Thus, we believe that a continuous improvement process is inevitable to achieve a good usability-reusability trade-off and thus an Information Model of high quality. In this work, we decided to compensate the lack of formal measures by applying DIM to two prototypical software applications:

- Application 1: Disassembly Sequence Generator
- Application 2: Adaptive Disassembly Planner based on the Disassembly Decision Network

For each of the application, an application layer sub model is developed based on the extension of the domain layer sub models and it contains more specific information required for the application.

Due to the page constraints in this paper, the detailed DIM structure and validation methodology will not be elaborated here. Readers are suggested to refer to Zhu B [36] for details.

Conclusion

This paper presents a Disassembly Planning Information Model, which constitutes a layered information framework designed for multiple applications in the domain of EOL product disassembly planning. Detailed requirements regarding the DIM are identified and analyzed in this paper, which looks into both of the domain aspect and modelling aspect.

The development of DIM has been presented very briefly in this paper. DIM is hierarchically structured by layers, which divide the associated Information Models into different levels of abstraction, and thus, separate the general knowledge from the specific knowledge about particular domains and applications. A set of submodels is thus developed and classified into three different layers named the abstract level, the domain level and the application level. DIM has been implemented into Web Ontology Language (OWL), which provides a foundation for the future IoT based product disassembly or recovery system.

References

1. Gharfalkar M, Court R, Campbell C, et al. (2015) Analysis of waste hierarchy in the European waste directive 2008/98/EC. Waste Management 39: 305-313.
2. Millar SA (2005) E-waste legislation. Paper, Film and Foil Converter 79: 16.
3. Dong J, Arndt G (2003) A review of current research on disassembly sequence generation and computer aided design for disassembly. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 217: 299-312.
4. Wang L, Wang XV, Gao L, et al. (2014) A cloud-based approach for WEEE remanufacturing. CIRP Annals 63: 409-412.
5. Xia K, Gao L, Wang L, et al. (2015) A semantic information services framework for sustainable WEEE management toward cloud-based remanufacturing. Journal of Manufacturing Science

- and Engineering 137.
6. Peneve KD, de Ron AJ (2007) Determination of a disassembly strategy. *International Journal of Production Research* 34: 495-506.
 7. Zhang HC, Kuo TC (1996) A graph-based approach to disassembly model for end-of-life product recycling. *Nineteenth IEEE/CPMT International Electronics Manufacturing Technology Symposium*, 247-254.
 8. Murayama T, Oba F, Abe S, et al. (2001) Disassembly sequence generation using information entropy and heuristics for component replacement. *Proceedings of the 2001 IEEE International Symposium on Assembly and Task Planning (ISATP2001). Assembly and Disassembly in the Twenty-first Century*, 208-213.
 9. Kanai S, Sasaki R, Kishinami T (1999) Graph-based information modeling of product-process interactions for disassembly and recycle planning. *Proceedings First International Symposium on Environmentally Conscious Design and Inverse Manufacturing*, 772-777.
 10. Lambert AJD (1997) Optimal disassembly of complex products. *International Journal of Production Research* 35: 2509-2524.
 11. Zussman E, Meng Chu Zhou, Caudill R (1998) Disassembly petri net approach to modeling and planning disassembly processes of electronic products. *Proceedings of the 1998 IEEE International Symposium on Electronics and the Environment. ISEE - 1998*, 331-336.
 12. Zussman E, Meng Chu Zhou (2000) Design and implementation of an adaptive process planner for disassembly processes. *IEEE Transactions on Robotics and Automation* 16: 171-179.
 13. Moore KE, Gungor A, Gupta SM (1998) A petri net approach to disassembly process planning. *Computers & Industrial Engineering* 35: 165-168.
 14. Chevron D, Binder Z, Horacek P, et al. (1996) Disassembling process modelling and operations planning under imprecise operation time. *IFAC Proceedings* 29: 5745-5750.
 15. Hsin-Hao H, Wang MH, Johnson MR (2000) Disassembly sequence generation using a neural network approach. *Journal of Manufacturing Systems* 19: 73-82.
 16. Yu B, Wu E, Chen C, et al. (2017) A general approach to optimize disassembly sequence planning based on disassembly network: A case study from automotive industry. *Advances in Production Engineering & Management* 12: 305-320.
 17. Ilgin MA (2016) Simultaneous determination of disassembly sequence and disassembly-to-order decisions using simulation optimization. *J Manuf Sci Eng* 138.
 18. Gao Y, Wang Q, Feng Y, et al. (2018) An energy-saving optimization method of dynamic scheduling for disassembly line. *Energies* 11: 1261.
 19. Lemaignan S, Siadat A, Dantan J, et al. (2006) MASON: A proposal for an ontology of manufacturing domain. *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications*, 195-200.
 20. Chang X (2008) Ontology development and utilization in product design.
 21. Kumar PP (2008) Design process modeling: Towards an ontology of engineering design activities.
 22. Kim K, Manley DG, Yang H (2006) Ontology-based assembly design and information sharing for collaborative product development. *Computer-Aided Design* 38: 1233-1250.
 23. Foufou S, Fenves SJ, Bock C, et al. (2005) A core product model for PLM with an illustrative XML implementation. *Inderscience Enterprises Limited, Geneve, Switzerland*, 21-32.
 24. Baysal MM, Roy U, Sudarsan R, et al. (2004) The open assembly model for the exchange of assembly and tolerance information: Overview and example. *ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 759-770.
 25. Feng SC, Kramer T, Sriram RD, et al. (2013) Disassembly process information model for remanufacturing. *J Comput Inf Sci Eng*, 13.
 26. Hamidullah, Bohez E, Irfan MA (2006) Assembly features: Definition, classification, and instantiation. *2006 International Conference on Emerging Technologies*.
 27. Kwaan JR (1994) Bayesian belief networks for industrial applications. *Proceedings of Adaptive Computing and Information Processing*, 625-640.
 28. (1983) 610.12-1990 - IEEE standard glossary of software engineering terminology. *Inst Electr & Electron Eng, USA*.
 29. Cysneiros LM, Werneck VM, Kushniruk A (2005) Reusable knowledge for satisficing usability requirements. *13th IEEE International Conference on Requirements Engineering*, 463-464.
 30. Borst P, Akkermans H, Top J (1997) Engineering ontologies. *International Journal of Human-Computer Studies* 46: 365-406.
 31. Klinker G, Bhola C, Dallemagne G, et al. (1991) Usable and reusable programming constructs. *Knowledge Acquisition* 3: 117-135.
 32. Gruber TR (1995) Toward principles for the design of ontologies used for knowledge sharing? *International Journal of Human-Computer Studies* 43: 907-928.
 33. Galster M, Avgeriou P (2012) A variability viewpoint for enterprise software systems. *2012 Joint Working Conference on Software Architecture & 6th European Conference on Software Architecture*, 267-271.
 34. Zhu B (2016) An information model in the domain of disassembly planning for sustainable manufacturing.
 35. Gennari JH, Musen MA, Fergerson RW, et al. (2003) The evolution of protege: An environment for knowledge-based systems development. *International Journal of Human-Computer Studies* 58: 89-123.
 36. Zhu B, Roy U (2018) Modeling and validation of a web ontology language based disassembly planning information model. *Journal of Computing and Information Science in Engineering* 18.

DOI: 10.36959/673/364