Journal of Engineering Design
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/cjen20

A high-definition design structure matrix (HDDSM) for the quantitative assessment of product architecture

Andrew Harold Tilstra [a] , Carolyn Conner Seepersad [b] & Kristin L. Wood [b]

[a] 3M, Software, Electronic, and Mechanical Systems Corporate Lab, 3M Center, St. Paul, MN, 55144, USA

[b] Department of Mechanical Engineering, The University of Texas at Austin, Austin, TX, 78712, USA

Version of record first published: 07 Aug 2012

To cite this article: Andrew Harold Tilstra, Carolyn Conner Seepersad & Kristin L. Wood (2012): A high-definition design structure matrix (HDDSM) for the quantitative assessment of product architecture, Journal of Engineering Design, DOI:10.1080/09544828.2012.706748

To link to this article: http://dx.doi.org/10.1080/09544828.2012.706748

Journal of Engineering Design
iFirst, 2012, 1–23

A high-definition design structure matrix (HDDSM) for the quantitative assessment of product architecture

Andrew Harold Tilstra[a], Carolyn Conner Seepersad[b]* and Kristin L. Wood[b]

[a]*3M, Software, Electronic, and Mechanical Systems Corporate Lab, 3M Center, St. Paul 55144, MN, USA;*
[b]*Department of Mechanical Engineering, The University of Texas at Austin, Austin 78712, TX, USA*

(*Received 24 November 2011; final version received 21 June 2012*)

As the field of engineering design matures, new techniques and methods are continuously being developed to conceptualise and analyse product architecture. These techniques and methods require product representations with higher sophistication, granularity, and fidelity. To address these needs, the high-definition design structure matrix (HDDSM) is presented as a new and evolved product representation model that captures a spectrum of interactions between components of a product, such that characteristics of product architecture can be assessed and compared. The HDDSM includes an interaction basis to capture a variety of standardised types of interactions and a hierarchical modelling method to facilitate modular, more efficient compilation of a design structure matrix with a high level of detail. To illustrate the types of quantitative analyses supported by the HDDSM, it is used as a foundation for quantifying the degree of nesting and identifying the presence of frameworks in product architectures – two characteristics that are related to product customisation.

Keywords: product architecture; functional modelling; product structuring; design structure matrix

1. Introduction

1.1. *Product architecture and component interactions*

The architecture of a product has a profound impact on its performance and the ease with which it can be changed or leveraged to accommodate different sets of requirements (Simpson 2004, Jiao *et al.* 2007). Product architecture is defined as the mapping of a product's functions to the physical components within the product and the interactions between those components (Ulrich 1995). Product architecture can be considered at various levels of detail, from systems to subsystems to individual components, and the interactions between those elements can take many forms, such as functional interactions involving transfer of energy and materials or spatial interactions involving physical proximity or contact. The types of interactions and their strength strongly influence the ease with which the product can be changed. Modular architectures, for example, often enable isolated changes to individual components, whereas integral architectures require

---

*Corresponding author. Email: ccseepersad@mail.utexas.edu

ISSN 0954-4828 print/ISSN 1466-1837 online

http://dx.doi.org/10.1080/09544828.2012.706748
http://www.tandfonline.com

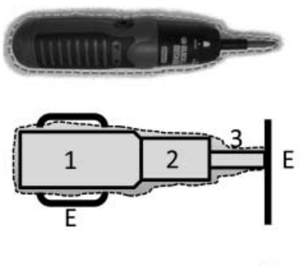
changes in multiple, highly coupled components to effect a change in functionality (Ulrich 1995). The relationship between a product's architecture and its ability to support variety and generational change is complex but potentially very beneficial, as evidenced by the abundance of recent research on product platforms, product families, and mass customisation (for reviews, see, Simpson 2004, Jiao *et al.* 2007). Models of product architecture are needed for reasoning about these complex relationships. The intent of this article is to develop a new and evolved model, based on design structure matrices, to capture and reason about complex product architecture.

1.2. *Design structure matrix*

Many researchers model product architecture with a component design structure matrix (DSM). A component DSM is used to capture and represent the interactions between elements of a system (Warfield 1973, Steward 1981, Browning 2001). An *element* may represent a single component, an assembly of components, or an abstract portion of the product system. The only requirement is that elements of the system be defined by non-overlapping boundaries. The elements of the system are used to label the rows and columns of a square matrix. A mark is placed in the matrix cell whenever two elements interact.

The basic concept of a DSM is introduced here with an abstract model of a battery-powered cordless screwdriver, as shown in Figure 1. The screwdriver is represented by three elements: a handle, a power source, and a bit. A fourth element is included to represent interactions between the elements and their external environment. The relationships can be recorded in a DSM as shown in the figure.

The example in Figure 1 provides a high-level introduction to a DSM, in which only the major subsystems of the product are represented. To effectively use this representation across a spectrum of design activities, increases in granularity (*Element Detail*) and fidelity (*Interaction and Judgment Detail*) are needed in the model. The level of *Element Detail* indicates the number of components modelled in the DSM and determines the dimensions of the square matrix as shown in Figure 2. This level can be quantified as the number of matrix elements used to represent the total system. The level of *Interaction Detail* indicates the types of interactions that can be defined and determines the length of the vector contained in each cell of the DSM. This level can also be thought of as creating layers of matrices (Figure 2), each containing one specific type of interaction. The level of *Judgment Detail* determines the type of variable contained in each individual cell. A Boolean variable is sufficient for recording the existence of an interaction, as in the cordless screwdriver example. The significance of an interaction requires a range of positive integers or ratios, and the consequence of an interaction requires both positive and negative



	1 Handle	2 Power Source	3 Bit	E External
Handle 1	▨	X		X
Power Source 2	X	▨	X	
Bit 3		X	▨	X
External E	X		X	▨

Figure 1. Abstraction and DSM of cordless screwdriver.

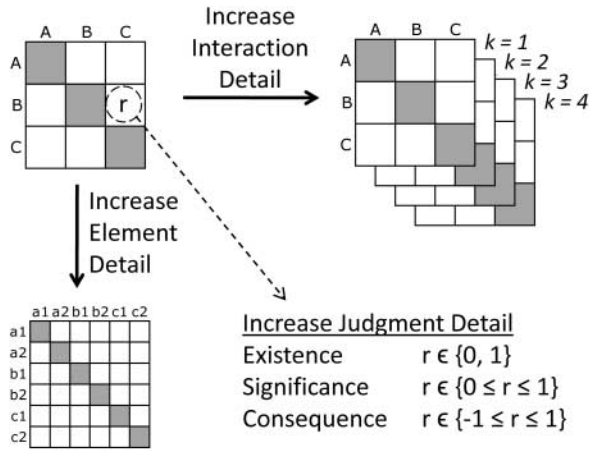


Figure 2. Element, Interaction, and Judgment Detail.

numbers. The Judgment Detail also includes key semantic information. The semantics can range from a binary existence rating to an interval scale representing the significance or likelihood of an interaction. Other semantic information can be documented such as the directionality or consequence of a relationship, represented by a positive or negative ‘sign’, or the probability of an interaction, represented with appropriate statistics.

1.3. *Related research*

There are many examples of component DSMs with varying levels of information content in the design literature. Pimmler and Eppinger (1994) use a component-based DSM to decompose and analyse an automatic climate control system. While Pimmler and Eppinger discuss the importance of selecting the elements that represent the system, there is no process for changing or amending the selected set of elements to effect an *a posteriori* change in the level of Element Detail in the DSM. Accordingly, it can be difficult and time-consuming to convert an abstract system- or subsystem-level model into a component-level model.

Pimmler and Eppinger (1994) consider four generic types of interactions between different components: energy, material, information, and spatial interactions. Sosa *et al.* (2003) extend the four generic interaction types used by Pimmler and Eppinger to include ‘structural’ as a fifth type of interaction that captures the transfer of mechanical support loads in their study of a jet engine. Helmer *et al.* (2010) provide an approach for aggregating these five types of interactions to efficiently identify clusters of components. While the five generic types of interactions offer more Interaction Detail than a simple binary assessment of overall interaction, a greater level of Interaction Detail is needed for comprehensively modelling the product architecture design space and for capturing component interactions in a standardised format that other researchers can easily interpret and replicate.

Luh *et al.* (2011) increase the Judgment Detail for the quantified design structural matrix with a variable between zero and one to indicate the ‘dependency strength’ of interactions between elements. Clarkson *et al.* (2004) use likelihood and impact ratings on the range of zero to one between components to better understand the risk of change propagation in a system. While the use of a DSM with increased Judgment Detail enhances the use of the model for a specific intended purpose, it may decrease the transferability of that DSM model for other uses. For large systems or for the comparison of many systems, additional training may be required to ensure that judgment ratings are used consistently.

An important role for a DSM model of product architecture is to support the study of product variety and product change, via analysis, benchmarking, and metric formulation. Research in change propagation, for example, is focused on the interconnection of elements in a DSM. Eckert *et al.* (2004) discuss the roles of elements as change multipliers, carriers, or absorbers in terms of change propagation. Suh *et al.* (2007) use a DSM-style 'change propagation matrix' to create a flexible platform for an automotive structural frame that can be used in different market segments. Martin and Ishii (2002) use a DSM-style 'coupling matrix' to study the specifications between the components in a water cooler product to manage variety in subsequent product offerings. Hsiao and Liu (2005) study the interactions between the parts in a drip coffee maker to manage the variety between different products in a product family. Alizon *et al.* (2007) stack the DSMs of variants in a product family to identify common interactions and modules. Luh *et al.* (2011) use a DSM of an existing product to optimise the design process and product architecture for a new family of power line communication products. Hölttä *et al.* (2005) use a DSM model of a product to generate a quantitative metric describing the modularity of the system and compare it with other measures of modularity.

While a component DSM provides valuable insight for the analysis and redesign of an embodied product, other similar tools may be more appropriate for other design tasks. For example, a multi-domain matrix supports management of the product development process by mapping dependencies between different types of system elements (e.g. components, people, and data), in a form that is similar to the hierarchical modelling strategy introduced in this article, but without the formalised interaction detail (Maurer and Lindemann 2008). Hubka, Andreasen, Ferreirinha, and colleagues created a chromosome model for supporting design; it hierarchically decomposes a system into functions, organs that carry the functions and behaviour, and 'wirk' structures and parts that embody those organs (cf. an overview by Andreasen (2011)). Like the work presented in this article, the chromosome approach supports hierarchically decomposing and modelling a product, but it lacks the structured DSM foundation and the standardised set of interactions presented in this article for populating it. Albers *et al.* (2011) introduce a Contact and Channel approach for modelling and managing interrelationships between function and form during an iterative design process, beginning with a rough concept. Like the work presented in this article, they build upon a standardised functional basis (Hirtz *et al.* 2002) and incorporate capabilities for hierarchically modelling systems at different levels of abstraction, but the work presented in this article utilises a DSM-based modelling approach, rather than a graphical approach, and focuses more strongly on analysing the architectures of existing products, which requires expanding the functional basis.

1.4. *A new and evolved product representation: the high-definition design structure matrix*

In this article, the high-definition design structure matrix (HDDSM) is introduced for modelling product architecture. As part of the HDDSM, interactions are categorised with an interaction basis that defines the types of interactions that may exist within a product architecture. Using this standardised set of interactions, the practitioner is required to assess only the existence of pre-defined interactions, rather than their degree of significance, thereby limiting the amount of judgment required of the practitioner and enhancing the repeatability of the model. An efficient method for constructing the HDDSM is also provided, which allows practitioners to increase (or decrease) the level of Element Detail with minimal effort and to construct a DSM in stages, from system-level models with low levels of Element Detail to component-level models with high levels of Element Detail.

With its high levels of standardisation and Interaction and Element Detail, the HDDSM supports the creation and implementation of a variety of metrics related to product change. Metrics are indispensible for quantifying important characteristics of product architecture, such as modularity, but they would be even more useful if they were derived from standardised product architecture models, such that the metrics can be compared consistently between different products and practitioners. In this article, two metrics are introduced for the purpose of illustrating exemplar analyses supported by the HDDSM.

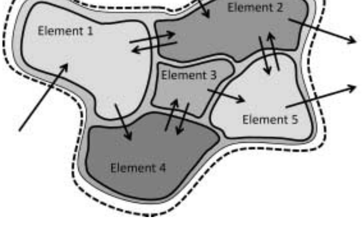
2. The high-definition design structure matrix

The HDDSM is a product architecture model that incorporates a modular modelling approach and a standardised interaction basis (Tilstra 2010). In Section 2.1, the modular modelling approach is presented, which allows different portions of a product or system to be modelled separately and then combined to create a larger model of the complete system. This procedure also allows a previously created model to be extended by modelling certain elements in more detail. In Section 2.2, a standard interaction basis is developed to scaffold the practitioner's assessment of different types of interactions within the product system. The standard interaction basis enables the development of consistent models by different practitioners, such that the models can be merged and compared with minimal explanation or subjective interpretation.

2.1. *A modular approach for constructing product architecture models*

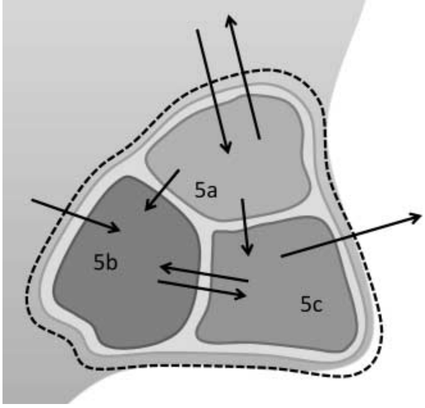
For the purpose of explaining the modular modelling approach, consider the abstract system shown in Figure 3. The system comprises a collection of elements that interact with each other, and certain elements may interact with the environment external to the system. The arrows in Figure 3 indicate the presence and direction of an interaction between the elements. This system can be represented in a DSM as shown. In Figure 4, the distinct system of Element 5 is shown with its corresponding DSM. In each of these systems, a model boundary is clearly defined by the dashed line. The external element, E, is used to represent interactions between system elements and the environment that is external to the system boundary. Each interaction is represented with a '1' in the matrix in which the column element is the source of the interaction, as indicated by the arrow direction. As shown in Figure 5, these two models are related by a hierarchical structure. The first model in Figure 3 uses five elements to represent the entire system, while the model in Figure 4 is a 'zoomed-in' view of just Element 5 and uses three elements to represent the subsystem.

A detailed model of the complete system, which includes all seven elements (1, 2, 3, 4, 5a, 5b, and 5c), can be populated primarily by the information already recorded in the matrices in Figures 3 and 4. Element pair interactions that are exactly the same as those from the two independent models are highlighted in dark grey with white text in Figure 6. The only 'new'



	1	2	3	4	5	E
1	-	1				1
2	1	-			1	1
3			-	1		
4	1		1	-		
5		1	1		-	
E		1			1	-

Figure 3. An abstract system that interacts with its environment and composed of elements.



	5a	5b	5c	E
5a	-			1
5b	1	-	1	1
5c	1	1	-	
E	1		1	-

Figure 4. Element 5 as a distinct system that interacts with its environment and composed of elements.

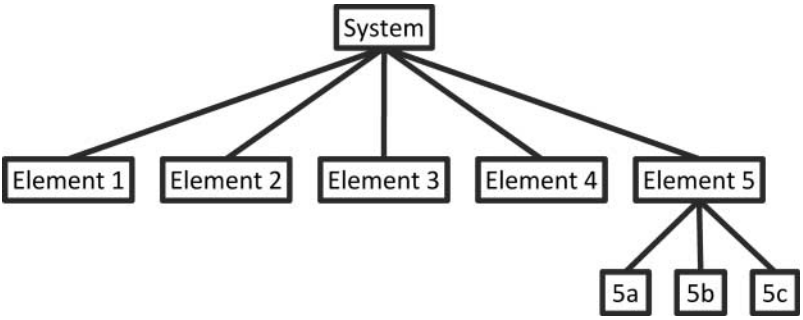
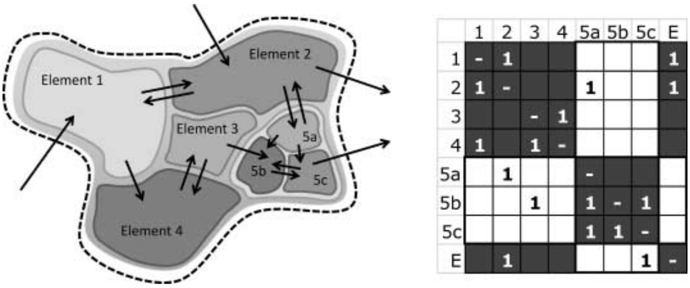


Figure 5. System hierarchy of elements.



	1	2	3	4	5a	5b	5c	E
1	-	1						1
2	1	-			1			1
3			-	1				
4	1		1	-				
5a		1			-			
5b			1		1	-	1	
5c					1	1	-	
E		1					1	-

Figure 6. The system represented by its smallest elements.

information required to build this more detailed model of the system is in the white areas of the matrix in Figure 6. For small systems, these unknown interactions can be quickly found by direct inspection of the system; as has been done in Figure 6. However, in larger engineering systems this may be difficult and, therefore, it is useful to focus the inspection effort of the examiner on areas of the system where potential interactions are likely.

Potential interactions between elements of the newly expanded system model can be identified by combining information from the individual system models in Figures 3 and 4. For the purpose of explanation, Figure 7 shows the matrix models of the independent systems organised into sub-matrices. Again, the grey areas represent information that can be directly used in the full, seven-element system model. In Figure 8, the sub-matrices are used to assemble the full system

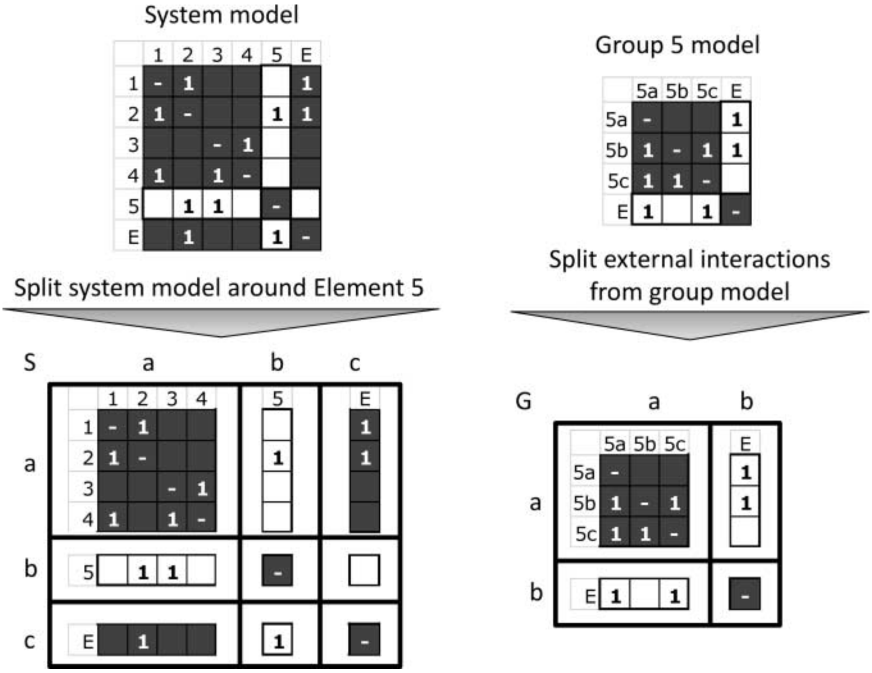


Figure 7. Independent models split into sub-matrices.

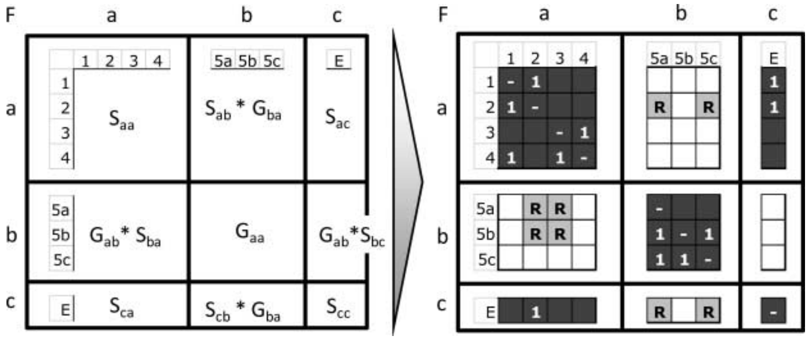
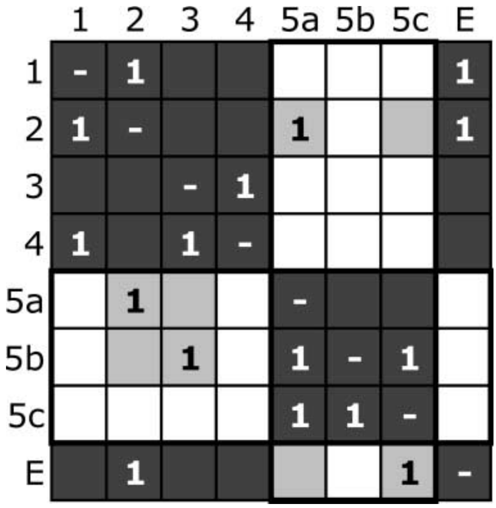


Figure 8. Predicting interactions between the group elements and the remaining system elements.

model. The sub-matrix F_{ab} represents interactions that occur between Elements 5a, 5b, and 5c and Elements 1, 2, 3, and 4 and originate in Elements 5a, 5b, and 5c. The sub-matrix F_{ab} is populated by multiplying S_{ab} (which represents interactions that occur between Element 5 and Elements 1, 2, 3, and 4 and originate in Element 5) with G_{ba} (which represents interactions between Elements 5a, 5b, and 5c and elements external to Element 5) as shown in Equation (1). The result of Equation (1) is shown in Figure 8 using '*R*' to indicate that these cells must be reviewed by the examiner. The other sub-matrices F_{ba}, F_{bc}, and F_{cb} can be similarly created by multiplication as indicated in the left matrix of Figure 8

$$F_{\text{ab}} = S_{\text{ab}} * G_{\text{ba}} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \qquad (1)$$



	1	2	3	4	5a	5b	5c	E
1	-	1						1
2	1	-			1			1
3			-	1				
4	1		1	-				
5a		1			-			
5b			1		1	-	1	
5c					1	1	-	
E		1					1	-

Figure 9. Final system model after reviewing.

An '*R*' in the cell of the system matrix in Figure 8 means that this element pair should be reviewed to check if an interaction actually exists. The remaining cells can be ignored because there are no possible interactions there. For example, there is no need to check the system model in Figure 6 to see if Element 5b is the source of an interaction with Elements 1, 2, 3, or 4 because it is known from the Group 5 model that Element 5b is not the source of any interactions outside of Group 5. The final system model is shown in Figure 9 again with the reviewed cells highlighted in light grey.

2.2. *The interaction basis*

The interaction layers of the HDDSM are defined by the interaction basis presented in Table 1. This interaction basis is a significant extension of the general interactions presented in other DSM-related research (Pimmler and Eppinger 1994, Helmer *et al.* 2010) and the flow set of the functional basis developed by Wood, Stone, and coauthors (Hirtz *et al.* 2002).

The functional basis is a common language for functional modelling. It represents a standard vocabulary of functional modelling terms that is intended to reduce ambiguity in functional models, facilitate comparison across functional models for different products, and increase uniformity of information across models created by different researchers (Hirtz *et al.* 2002). The primary and secondary classes of the flow set define specific interactions between elements in a functional model. At the primary level, the flows in functional modelling are signal, material, and energy. The secondary class of the flow set provides a more detailed list of these types of flows as represented by flows such as thermal energy and liquid material. The primary and secondary classes of the flow set have been incorporated into the general and specific interactions, respectively, listed in Table 1, and Hirtz *et al.* (2002) offer a complete description of those terms.

The flow set from the functional basis is not sufficient, however, to describe the interactions between physical products because functional models are intentionally created independent of product form. Accordingly, the interaction basis in Table 1 has been expanded to include additional interactions. For example, Pimmler and Eppinger (1994) define a spatial-type interaction as the need 'for adjacency or orientation between two elements'. In Table 1, 'spatial' is included as a general type of interaction, with specific interactions of the spatial type including proximity and alignment. Proximity, like adjacency, indicates that two elements are physically close to

Table 1. Interaction basis for HDDSM.

General	Specific	Abbreviation
Information[a,b]	Status[a]	SI
	Control[a]	CI
Material[a,b]	Human[a]	HM
	Gas[a]	GM
	Liquid[a]	LM
	Solid[a]	SM
	Plasma[a]	PM
	Mixture[a]	MM
Energy[a,b]	Human[a]	HE
	Acoustic[a]	AE
	Biological[a]	BE
	Chemical[a]	CE
	Electrical[a]	EE
	Electromagnetic[a]	EME
	Hydraulic[a]	HYE
	Mechanical[a]	ME
	Magnetic[a]	MAG
	Pneumatic[a]	PE
	Radioactive[a]	NE
	Thermal[a]	TE
	Strain energy	SE
Spatial[b]	Proximity	P
	Alignment	A
Movement	Translational	LRM
	Rotational	RRM

[a] Used in flow set of functional basis (Hirtz *et al.* 2002).
[b] Used in related DSM research (such as Pimmler and Eppinger 1994, Helmer *et al.* 2010).

one another, such that neither element can be enlarged without contacting the other. An alignment interaction means that an element determines the location, orientation, or path of the corresponding element.

Along with proximity, strain energy interactions are used to represent structural contact. Two components that are in structural contact deform one another as they transfer structural loads between them. The work required to induce these deformations is defined as strain energy, which can also be viewed as the potential energy stored in two contacting components as a result of the deformation. Strain energy can be distinguished from mechanical energy as a distinct form of interaction in the interaction basis. Mechanical energy involves a transfer of energy between moving components, such as the transfer of torque between the output shaft of a motor and a transmission, or the transfer of translational energy between an expanding spring and a projectile. Strain energy typically involves the potential energy stored in the form of deformation between two components that are in contact for the purpose of transferring static, structural loads, such as the two halves of a casing that are joined by screws or snap fits. While *mechanical energy* is typically recorded as an asymmetric interaction to capture the directional flow of mechanical energy within the system, *strain energy* is typically recorded as a symmetrical interaction to represent the static nature of the underlying structural connections. Strain energy can be helpful in identifying elements that are nested or unnecessarily close to one another, for example, because those elements will exhibit spatial proximity interactions without strain energy interactions.

Movement between components is also an important type of interaction. The specific types of movement may be translational relative movement or rotational relative movement. When a component moves along a linear path relative to another component, it is providing a translational, or linear, relative motion interaction. When a component spins within, spins around, or orbits about another component, it is providing an interaction of rotational relative motion. Although movement has not been included as a type of interaction in related research, it is found to be important in this research for better understanding the complete interactions between components. For example, the gears in a transmission are 'connected' in a sense, but that connection is much different from a fixed structural connection between two beams because the gears are in motion.

3. HDDSM modelling

An HDDSM model can be created as part of a reverse engineering process that starts with a working product, proceeds through product teardown and disassembly, and finishes with a complete model of all the interactions between all parts of the product. An overview of the process for creating the HDDSM for a Black and Decker® power screwdriver is provided in this section. The power screwdriver shown in Figure 10 is a consumer product that is of sufficient complexity to explain the modelling procedure.

The overall steps of the process are listed below:

(1) Reverse engineer the product.
(2) Assign parts to groups.
(3) Create system-level HDDSM.
(4) Create group-level HDDSMs.
(5) Merge group-level HDDSMs into the system-level HDDSM.
(6) Utilise HDDSM for product analysis.

Figure 10. Black and Decker® power screwdriver.

3.1. *Reverse engineer the product*

The first step in any reverse engineering method is to fully understand the product being examined, using a variety of tools, including an activity diagram and a Bill of Materials (BOM) (Otto and Wood 2001). When building the HDDSM, it is important to focus on the primary activity within the product's activity diagram, such as the activity of driving screws for the cordless screwdriver. It is also important to compile or obtain a BOM for the product, as shown in Table 2 for the cordless screwdriver. When creating the BOM, it is acceptable to include parts, such as original equipment manufacturer parts, that are available as a single unit, even though they comprise a number of unique parts. These preassembled parts can be decomposed subsequently into constituent elements, as discussed in Section 2.1.[1]

3.2. *Assign parts to groups*

After the product has been disassembled and each part has been identified, the parts of the product must be allocated into groups to be modelled separately. A 'part' is defined as a single unique entity on the BOM. An 'element' is the single unique entity of a particular HDDSM model, which can represent a single part or a collection of parts. Since the HDDSM is a matrix-based product representation, the number of possible element pair interactions to be considered by the person examining the system is determined by the number of elements defined for the system. The number

Table 2. BOM for B&D power screwdriver.

	Part	Group name	Quantity	Manufacturing process	Material
1	Battery holder	Battery holder	1	Injection moulded	Plastic
2	AA battery	Battery holder	4	Standard part	Various
3	Series terminal clip	Battery holder	2	Stamped and bent	Steel
4	Battery plug	Battery holder	1	Injection moulded	Plastic
5	Plug terminals	Battery holder	2	Stamped and bent	Steel
6	Series terminal plate	Battery holder	1	Stamped and bent	Steel
7	Bottom housing	Battery holder	1	Injection moulded	Plastic
8	Bit	Bit	1	Standard part	Metal
9	Negative terminal clip	Electrical system	1	Stamped and bent	Steel
10	Positive terminal clip	Electrical system	1	Stamped and bent	Steel
11	Terminal carrier	Electrical system	1	Injection moulded	Plastic
12	Button	Electrical system	1	Injection moulded	Plastic
13	Front housing	Housing	1	Injection moulded	Plastic
14	Back housing	Housing	1	Injection moulded	Plastic
15	Pin	Housing	2	Standard part	Metal
16	Motor	Motor	1	Standard part	Various
17	Clip	Transmission	1	Bent rod	Steel
18	Motor gear	Transmission	1	Standard part	Metal
19	Washer	Transmission	1	Standard part	Steel
20	Planetary gears, Set1	Transmission	3	Injection moulded	Plastic
21	Carrier sun	Transmission	1	Machined	Steel
22	Planetary gears, Set2	Transmission	3	Injection moulded	Plastic
23	Planetary carrier	Transmission	1	Machined	Steel
24	Ring gear, housing	Transmission	1	Injection moulded	Plastic
25	Ring clip, shaft	Transmission	1	Standard part	Metal
26	Shaft	Transmission	1	Machined	Steel
27	Brake gear	Transmission	1	Injection moulded	Plastic
28	Chuck	Transmission	1	Machined	Steel
29	Brake switch	Transmission	1	Injection moulded	Plastic
30	Brake switch spring	Transmission	1	Stamped and bent	Steel
31	Ring clip	Transmission	1	Standard part	Metal
32	Washer, ring clip	Transmission	1	Standard part	Metal
Total quantity of components			42		

of element pairs, or cells in the matrix, to be examined can be calculated by Equation (2)

$$\text{Element pairs} = n^2 - n, \tag{2}$$

where n is the total number of elements in a particular HDDSM model.

For any given product, there are many ways to cluster parts into groups. Based on the judgment of the examiner, parts could be grouped according to functional flows or proximity, for example. The goal when assigning parts to groups is to recognise sets of parts that form a highly interconnected group so that the practitioner's effort is focused on recognising the specific types of interactions rather than considering element pairs that have no potential for interaction. For example, when disassembling the power screwdriver it was noticed that the internal gears of the transmission are completely encapsulated in the transmission ring gear and cannot interact with the battery holder. By assigning the internal gears of the transmission to a group that is independent from the parts of the removable battery holder, the examiner will, for example, not spend any time evaluating whether the planetary gears interact with the battery.

For the Black and Decker® power screwdriver, the system-level HDDSM is defined by grouping the parts of the BOM into six different elements. A column in the BOM spreadsheet, in Table 2, is designated to record the element group assignment of each unique part. This is done to ensure that parts listed in the BOM are assigned to only one element in the system-level HDDSM. The six groups selected are listed below along with the 'External' element used to capture interactions with the external surroundings:

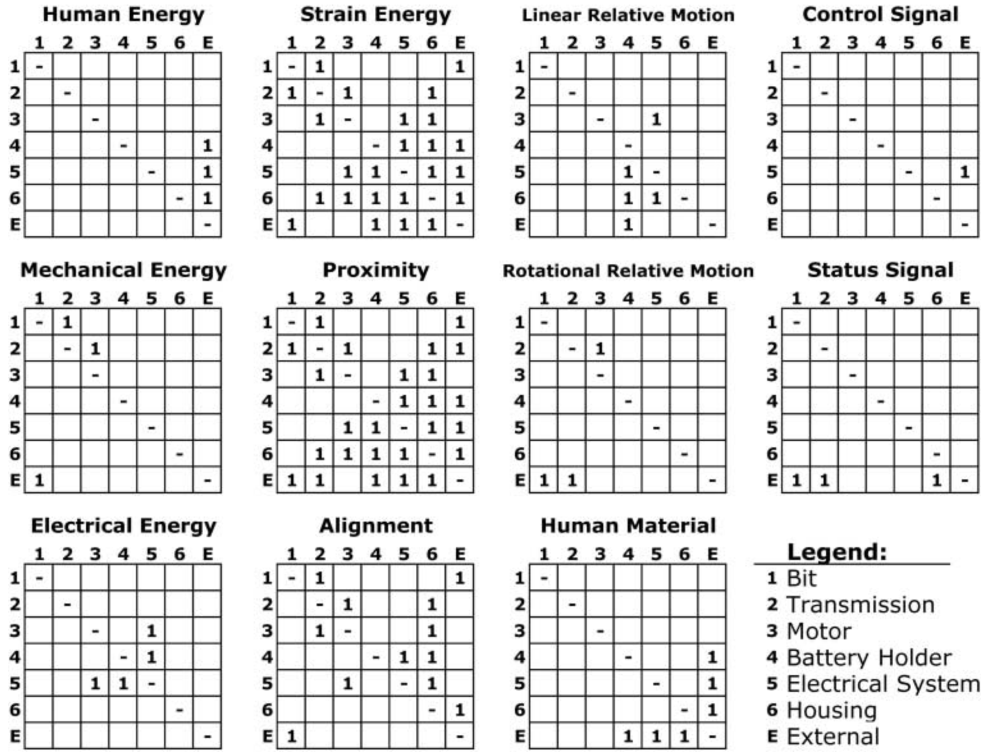
(1) Bit.
(2) Transmission.
(3) Motor.
(4) Electrical system.
(5) Battery holder.
(6) Housing.
(7) External.

The BOM created during disassembly indicated a total quantity of 42 parts. Since this system-level HDDSM uses 6 elements to represent all 42 parts, it is at 14% relative element detail ($6/42 = 0.14$). As the subsystems are modelled and expanded, the percent of relative element detail is increased.

3.3. *Create the system-level HDDSM*

The system-level HDDSM is created by evaluating every potential pairing of group elements and recording the specific type and direction of interactions between them. The number of pairs to examine grows quadratically with the number of elements included in the HDDSM according to Equation (2). If there are 10 elements in the HDDSM model, then there are 90 pairs to examine. If there are 20 elements in the HDDSM model, then there are 380 pairs to examine. Based on prior experience, it can be difficult to create a HDDSM greater than 10 elements without using some form of information management software to manage and manipulate the data reliably. The HDDSM Manager software has been written for the MATLAB platform to serve this purpose (Tilstra 2010). The HDDSM Manager software facilitates the process of examining element pairs by guiding the examiner through all of the possible pairs.

In the Black and Decker® screwdriver, 11 of the 25 specific interactions listed in Table 1 are identified between the groups. The remaining 14 layers of interaction are unused for this product. For each of the identified interaction types, a DSM matrix is created which constitutes one 'layer' of the HDDSM as shown in Figure 11. For example, in the 'mechanical energy layer' the cell



Human Energy

	1	2	3	4	5	6	E
1	-						
2		-					
3			-				
4				-			1
5					-		1
6						-	1
E							-

Strain Energy

	1	2	3	4	5	6	E
1	-	1					1
2	1	-	1			1	
3		1	-		1	1	
4				-	1	1	1
5			1	1	-	1	1
6		1	1	1	1	-	1
E	1			1	1	1	-

Linear Relative Motion

	1	2	3	4	5	6	E
1	-						
2		-					
3			-		1		
4				-			
5				1	-		
6				1	1	-	
E				1			-

Control Signal

	1	2	3	4	5	6	E
1	-						
2		-					
3			-				
4				-			
5					-		1
6						-	
E							-

Mechanical Energy

	1	2	3	4	5	6	E
1	-	1					
2		-	1				
3			-				
4				-			
5					-		
6						-	
E	1						-

Proximity

	1	2	3	4	5	6	E
1	-	1					1
2	1	-	1			1	1
3		1	-		1	1	
4				-	1	1	1
5			1	1	-	1	1
6		1	1	1	1	-	1
E	1	1		1	1	1	-

Rotational Relative Motion

	1	2	3	4	5	6	E
1	-						
2		-	1				
3			-				
4				-			
5					-		
6						-	
E	1	1					-

Status Signal

	1	2	3	4	5	6	E
1	-						
2		-					
3			-				
4				-			
5					-		
6						-	
E	1	1				1	-

Electrical Energy

	1	2	3	4	5	6	E
1	-						
2		-					
3			-		1		
4				-	1		
5			1	1	-		
6						-	
E							-

Alignment

	1	2	3	4	5	6	E
1	-	1					1
2		-	1			1	
3		1	-			1	
4				-	1	1	
5			1		-	1	
6						-	1
E	1						-

Human Material

	1	2	3	4	5	6	E
1	-						
2		-					
3			-				
4				-			1
5					-		1
6						-	1
E				1	1	1	-

Legend:

1 Bit
2 Transmission
3 Motor
4 Battery Holder
5 Electrical System
6 Housing
E External

Figure 11. Data recorded in system-level HDDSM of power screwdriver.

in the first row and second column is marked with a ‘1’, indicating that the ‘transmission’ group (Element 2) provides mechanical energy to the ‘Bit’ (Element 1).

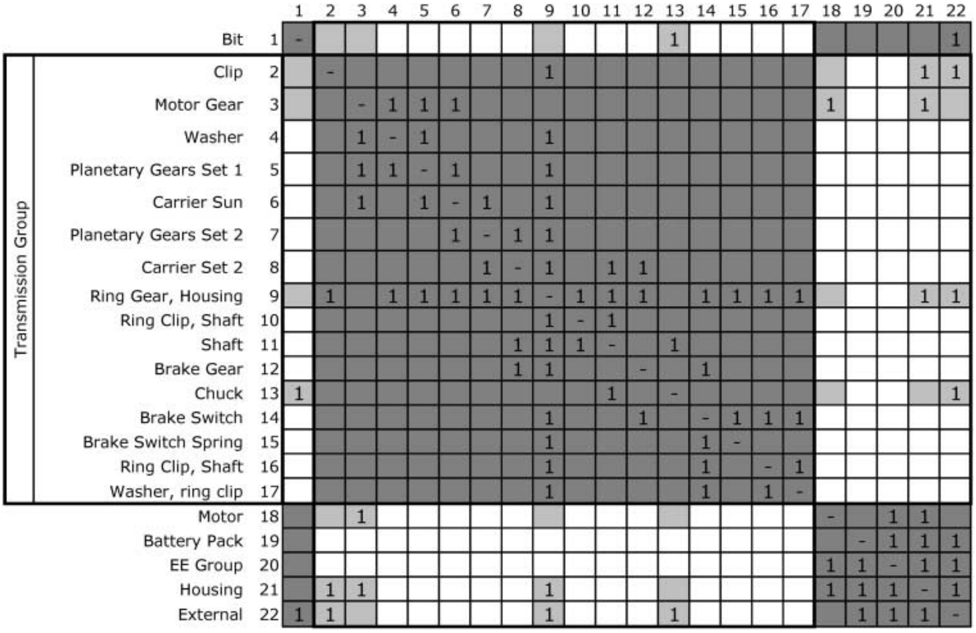
3.4. *Create group-level HDDSMs*

A HDDSM model for each group defined in Table 2 could be created independently. Each of these group HDDSMs would contain its own ‘external’ element to capture interactions that cross the group’s boundary. For example, in Table 2, the ‘motor gear’ is allocated to the transmission group. In the transmission group HDDSM, the ‘motor gear’ would receive mechanical energy from something external to the transmission. Later, when the transmission group is merged into the system-level HDDSM model, the practitioner would be prompted to review if the ‘motor gear’ is receiving this mechanical energy specifically from the motor.

The inclusion of the external element is a powerful concept because it allows the transferability of group, or subsystem, HDDSM models into other systems. Therefore, if a module is common across a large range of products, this module can be modelled once, and the module’s HDDSM can be reused for each of the individual product HDDSMs.

3.5. *Merge group-level HDDSMs into the system-level HDDSM*

The group-level HDDSMs may be merged into the system-level HDDSM using the process described in Section 2.1 for every type of interaction. Figure 12 shows the results of merging the transmission group into the system-level HDDSM for the proximity interactions only. The dark grey areas with white text represent information that is not affected during the merge. The light



			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
	Bit	1	-												1									1
Transmission Group	Clip	2		-							1												1	1
	Motor Gear	3			-	1	1	1												1			1	
	Washer	4			1	-	1				1													
	Planetary Gears Set 1	5			1	1	-	1			1													
	Carrier Sun	6			1		1	-	1		1													
	Planetary Gears Set 2	7						1	-	1	1													
	Carrier Set 2	8							1	-	1		1	1										
	Ring Gear, Housing	9		1		1	1	1	1	1	-	1	1	1		1	1	1	1				1	1
	Ring Clip, Shaft	10									1	-	1											
	Shaft	11								1	1	1	-		1									
	Brake Gear	12								1	1			-		1								
	Chuck	13	1										1		-									1
	Brake Switch	14									1			1		-	1	1	1					
	Brake Switch Spring	15									1					1	-							
	Ring Clip, Shaft	16									1					1		-	1					
	Washer, ring clip	17									1					1		1	-					
	Motor	18			1															-		1	1	
	Battery Pack	19																			-	1	1	1
	EE Group	20																		1	1	-	1	1
	Housing	21		1	1						1									1	1	1	-	1
	External	22	1	1							1				1						1	1	1	-

Figure 12. Merging the transmission group into the system-level proximity interactions.

grey cells represent element pairs that have potential interaction and need to be reviewed. The white cells represent element pairs that are predicted to have no interaction based on information previously entered in the HDDSM models. These white cells do not need to be reviewed by the practitioner. In this example, there are 16 element pairs that need to be reviewed. The HDDSM merging process reduces the effort of the practitioner by excluding additional 80 element pairs from further evaluation. This process has been automated using the HDDSM Manager software described by Tilstra (2010).

The practitioner examining the system can review the possible new system interactions to determine which interactions truly exist in the product. Examination of the product would confirm that the ‘Clip (2)’ element connects the ‘ring gear, Housing (9)’ element to the ‘Housing (21)’ element and is indeed in proximity. It is also indicated in Figure 12 that the ‘Clip (2)’ could be proximal to the ‘Motor (18)’ element. However, examination of the product shows this not to be true.

The author created a HDDSM for the Black and Decker® power screwdriver at 76% Element Detail, meaning that each of the 32 parts on the BOM has a corresponding element in the final HDDSM. (Total quantity of parts is 42 when duplicate parts are counted.) Using the HDDSM Manager software described in Tilstra (2010), this task was performed in less than a day. The process of inferring and reviewing possible interactions through merging subsystems reduced the overall number of element pairs to be manually evaluated by 52%. The hierarchical modelling approach is intended to effect similar reductions in the practitioner’s modelling effort for other products, as well, but it is difficult to quantify the effect *a priori*. The magnitude of the effort reduction depends on the size of the system, the inherent modularity of the system, and the manner in which the practitioner groups the components.

3.6. *Utilise HDDSM for product analysis*

The HDDSM process offers a strategic approach to collecting product architecture data. Once collected, these data can be used for a range of different types of analysis. The different types of interaction data recorded in the HDDSM can be used separately or combined for different types of analysis. In the next section, the HDDSM is used to support two representative types of quantitative analysis of product architecture.

4. Examples of quantitative analysis of product architecture supported by the HDDSM

Several aspects of product architecture have a significant influence on the ease with which a product can be redesigned to meet changing requirements. In this section, two of those characteristics – space potential and frameworks – are described as motivating examples for the types of quantitative analysis that can be supported by the HDDSM. In this section, the analyses are applied to the consumer products pictured in Figure 13; those products are selected for their (moderate) complexity and for the noticeable variety of architectures that they embody. The purpose of analysing a variety of products, rather than a line of functionally similar products, is not to directly compare the products to one another, *per se*, but to demonstrate how the HDDSM can support quantitative analysis of a variety of different product architectures and identification of (un)desirable characteristics of each individual architecture, such as component nesting or the existence of a structural framework, without necessarily requiring functionally similar comparison products.

4.1. *Analysis of space potential using the HDDSM*

Open space, or ‘headroom’, within a product architecture can be desirable if it facilitates expansion of individual components without requiring rearrangement or repositioning of neighbouring components (Martin and Ishii 2002, Kuchinsky 2005, Qureshi *et al.* 2006, Keese *et al.* 2007, Otto and Holtta-Otto 2007, Tilstra 2010), and it may help avoid the negative effects of nesting (Ulrich 1995) and provide access to components for assembly/disassembly (Boothroyd *et al.* 2002).

The prevalence of ‘headroom’ or open space within a specific product architecture can be quantitatively analysed based on the HDDSM. Since interactions of proximity are recorded separately in the HDDSM, element pairs in the HDDSM that have interactions of proximity *without* the other interaction types are identified as element pairs with ‘space potential’ or the potential for adding more expansion space. Element pairs with space potential can be filtered from the HDDSM using

Figure 13. Products used in study of product architecture metrics.

simple Boolean algebra or cell-by-cell multiplication of the appropriate interaction matrices. To compare alternative designs or products, it is useful to normalise the number of element pairs with proximity as the only interaction mode by the total number of element pairs with proximity interactions. The result is the space potential ratio (SPR) as shown in Equation (3).

$$\mathrm{SPR} = \frac{N_{\mathrm{SP}}}{N_{\mathrm{P}}}, \tag{3}$$

where SPR is the space potential ratio, N_{SP} the number of element pairs with proximity as the only interaction mode, and N_{P} the number of element pairs with proximity interactions.

Values of the SPR metric range from zero to one. A low SPR metric is desirable because it indicates that parts are only located near each other if they functionally need to be. A high SPR metric is undesirable because it suggests that many parts are in proximity to each other but do not necessarily need to be for functional reasons. The calculation of this metric would not be possible without the detailed functional interactions captured in the HDDSM.

The SPR was calculated for the variety of consumer products illustrated in Figure 13, and the results are shown in Figure 14. The Cooper Wine Bottle Cooler™, for example, has a high SPR. As shown in Figure 15, the circuit board, motors, and plumbing in the Cooper Cooler™ are tightly packed together, making it difficult to resize, reshape, or reorient any of the components without making changes to neighbouring components, as well. Similarly, in the Black and Decker® Jar

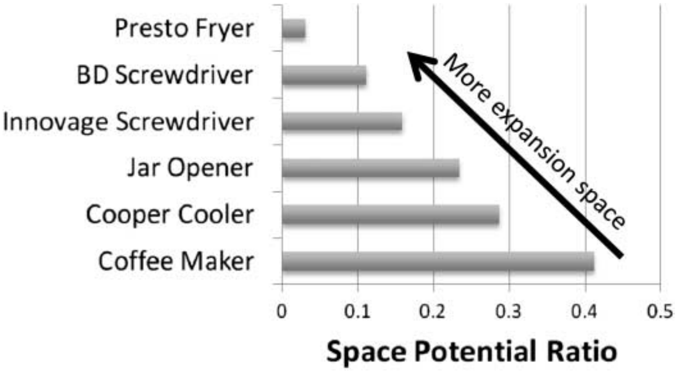


Figure 14. SPR for a variety of consumer products.

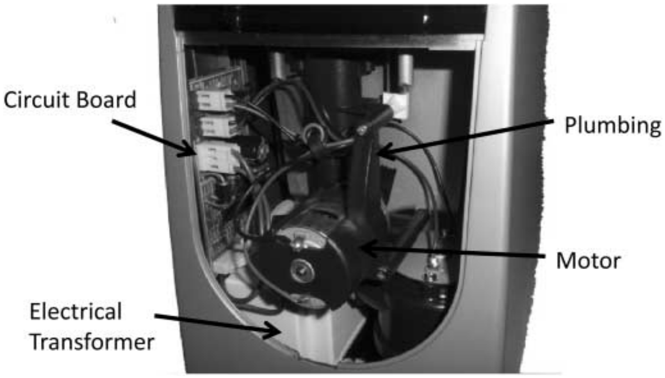


Figure 15. Circuits, motors, and plumbing inside the Cooper Cooler™.

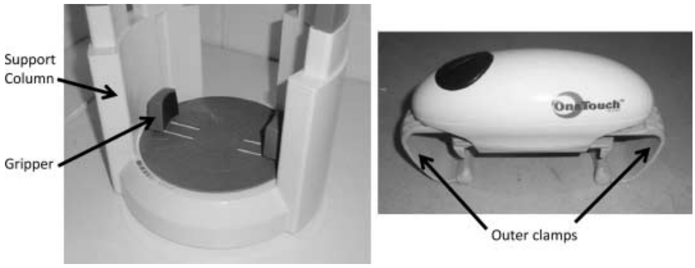


Figure 16. Black and Decker® Jar Opener (left) and OneTouch™ Jar Opener (right).

Opener, the proximity of the support columns to the clamping mechanism creates an unnecessary limitation on the diameter of the clamping mechanism (Figure 16). Therefore, relative to alternative product architectures in the market, it would be difficult to evolve this design into a future product capable of opening larger jars. For example, the OneTouch™ jar opener, also shown, in Figure 16 could be evolved to open larger jars by redesigning only the two outer clamping parts.

4.2. *Identification of product frameworks using the HDDSM*

'Bus' architectures, which establish one or more components as a framework for other components, make it easier for a product architecture to support design changes, by minimising the propagation of those changes to other components (Rothwell and Gardiner 1988, Ulrich 1995, Fricke and Shulz 2005, Qureshi *et al.* 2006, Keese *et al.* 2007, Tilstra 2010). In mechanical products, frameworks are typically used for 'mounting' modules, implying that a framework will have large numbers of structural connections to neighbouring components. In an HDDSM, structural connections can be identified as interactions of strain energy, spatial proximity, and spatial alignment. An element with large numbers of structural connections, relative to other elements in the product, is likely to be a framework.

A Framework metric is defined in terms of the number of structural interactions, y, provided by each element in the HDDSM. The Framework metric is calculated by identifying the number of structural interactions provided by each element in the HDDSM, y; rank ordering the elements in terms of the number of structural interactions; identifying the element with the largest number of structural interactions, $y_{\max}$; calculating the difference in the number of structural interactions between any two consecutive elements in a ranked list of structural interactions, Δy; identifying the maximum Δy; and then calculating the Framework metric as follows:

$$\mathrm{FW} = \frac{(\Delta y)_{\max}}{y_{\max}}, \tag{4}$$

where FW is the Framework metric, $(\Delta y)_{\max}$ the maximum difference in the number of structural interactions between two consecutive elements in a ranked list of structural interactions, and $y_{\max}$ the maximum number of structural interactions provided by any element in the architecture.

Larger values of the Framework metric (near one) indicate more prominent frameworks. A Framework metric closer to zero indicates that there is a gradual change in the number of structural interactions provided by each element and, therefore, a distinct framework is not evident in the design. The metric identifies frameworks that are similar to the integrative components defined by Sosa *et al.* (2003) as components that connect with numerous components across the system.[2]

As an example, consider the three theoretical systems shown graphically in Figure 17. The DSMs for the systems are shown in Figure 18. The number of structural interactions, y, provided from each element in the system is counted by adding down the columns of the DSMs in Figure 18.

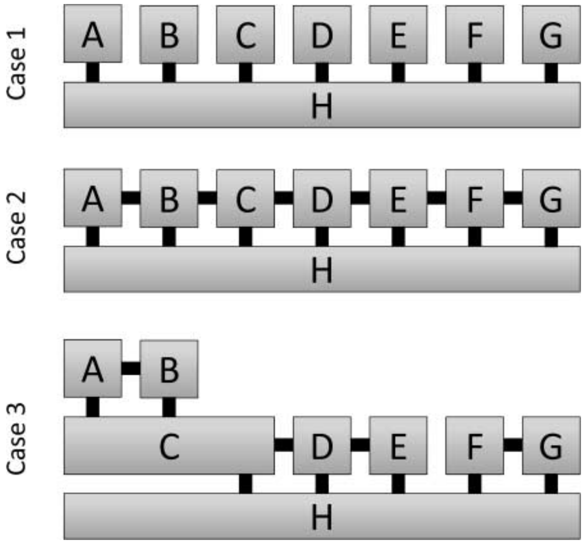
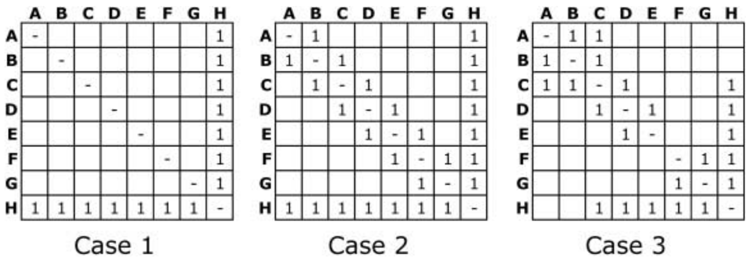


Figure 17. Theoretical exhibitions of systems with a framework.



	A	B	C	D	E	F	G	H
A	-							1
B		-						1
C			-					1
D				-				1
E					-			1
F						-		1
G							-	1
H	1	1	1	1	1	1	1	-

Case 1

	A	B	C	D	E	F	G	H
A	-	1						1
B	1	-	1					1
C		1	-	1				1
D			1	-	1			1
E				1	-	1		1
F					1	-	1	1
G						1	-	1
H	1	1	1	1	1	1	1	-

Case 2

	A	B	C	D	E	F	G	H
A	-	1	1					
B	1	-	1					
C	1	1	-	1				1
D			1	-	1			1
E				1	-			1
F						-	1	1
G						1	-	1
H			1	1	1	1	1	-

Case 3

Figure 18. DSMs for systems with a framework.

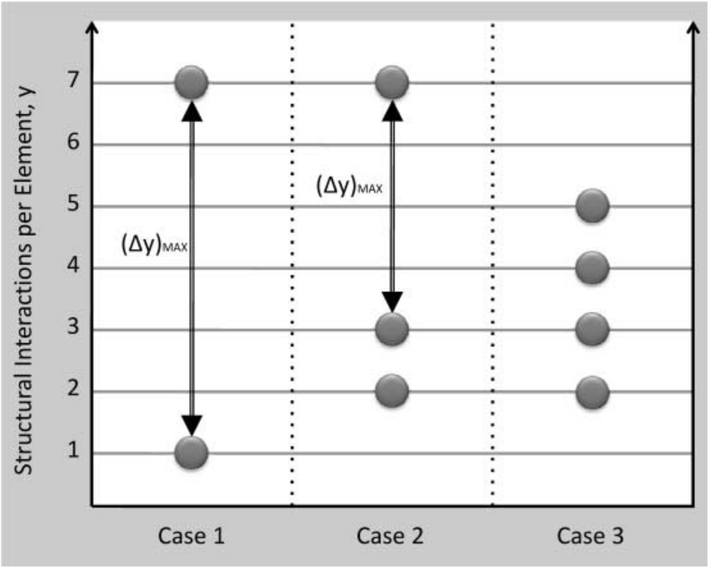


Figure 19. Structural interactions chart.

For each system being studied, markers are placed on the chart in Figure 19 for each element based on its number of structural interactions. Many elements will have the same number of structural interactions, so a single marker may represent several elements with an equivalent number of structural interactions.

From Figure 19, it is clear that in Cases 1 and 2 the framework is easily distinguished as a component with a significantly greater number of structural connections than neighbouring elements. Using Equation (4), the Framework metric for Case 1 is $6/7 = 0.86$ and for Case 2 it is $4/7 = 0.57$. However, in Case 3 the distinction is not as significant; since there is no single, maximum difference between the markers in the structural interaction chart, there are no clear frameworks in the system.

Figure 20 shows the structural interaction chart of the products in Figure 13. Vertical lines represent $(\Delta y)_{\max}$ for each product. Since all of the product models have a different number of elements and a different range of structural connections, structural interactions are normalised by the maximum number of structural interactions in each product, to facilitate plotting all products on the same chart. As expected from the theoretical examples above, the products have a similar profile, with a relatively small number of elements that are highly connected. The number of connections then quickly declines, and a majority of the elements have much smaller numbers of structural interactions.

Potential frameworks can be visually identified as any elements ranked above the vertical lines in Figure 20. Figure 21 plots values of the Framework metric for the products illustrated in Figure 13. The Cooper Cooler™ has one element, the ‘main body’, that provides significantly more structural interactions to other elements in the system (Figure 22). The Framework metric determined by Equation (4) is equal to 0.63. The Black and Decker® Jar Opener has a much smaller metric of

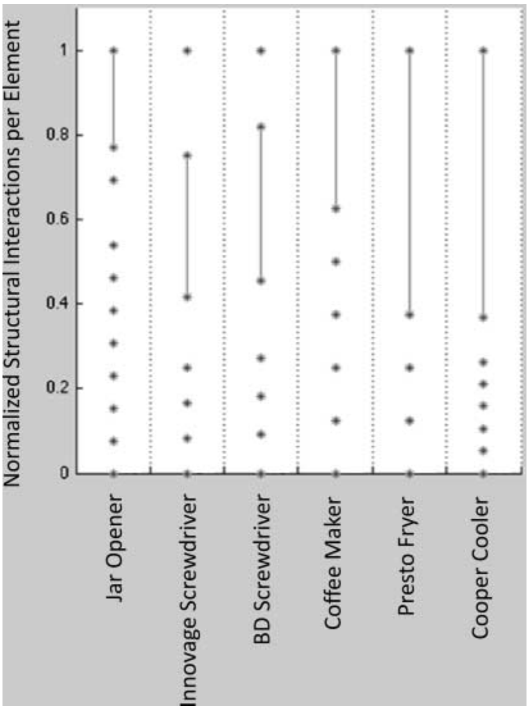


Figure 20. Normalised structural interaction chart.

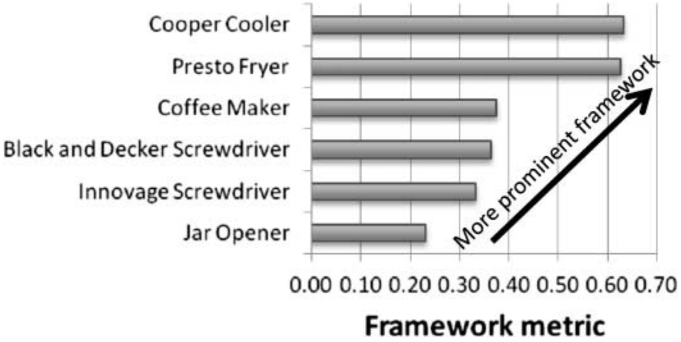


Figure 21. Framework metric.

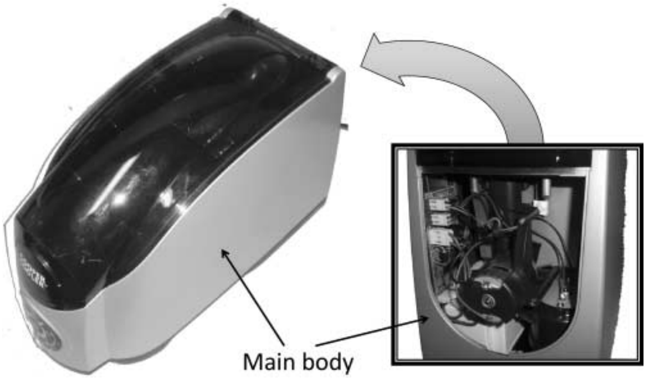


Figure 22. Cooper Cooler™, main body element.

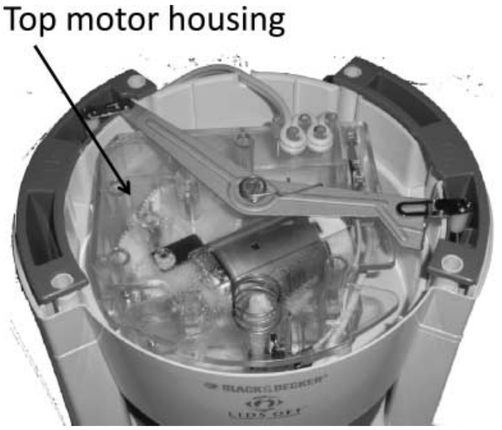


Figure 23. Jar Opener, Top motor housing element.

0.23. From Figure 20, it can be seen that the maximum Δy is less pronounced for the jar opener. When reverse engineering this product, it is evident that the 'top motor housing' element shown in Figure 23 is a structurally important component; however, examining the design with a quantitative metric allows the practitioner to focus on *how much* more connected it is, relative to other parts.

4.3. *Benefits of multiple quantitative metrics*

These examples demonstrate how the HDDSM can be used to support quantitative analysis of product architectures. An interesting aspect of HDDSM-based architectural analysis is that the same HDDSM model for a product can be used to support different analyses and that these analyses can be used to highlight relative strengths and weaknesses of various architectures.

For example, the Presto® Fryer and the Cooper Cooler™ are typical kitchen small appliances. Although the two products are manufactured using different methods, they both have Framework metric values above 0.6, which indicates that some components in each design are significantly more structurally important than other components. The Cooper Cooler™ has a large injection moulded, plastic body onto which the other components are attached. The Presto® Fryer has a cast aluminium bowl onto which the other components are attached. From this single perspective, the product architecture of these two products is very similar.

When the SPR for the Presto® Fryer is compared with that of the Cooper Cooler™, the differences between the product architectures are very clear. The components of the Presto® Fryer are attached on the outside of the framework and they do not interact with each other. For example, vast design changes could be made to the size and shape of the handles on the fryer bowl without those changes propagating to the feet or the lid. The components of the Cooper Cooler™ are nested inside the framework and it is likely that changes to some of the components would necessitate changes in others.

While these two products may be simplistic compared with larger, more complex systems, the example demonstrates the benefit of being able to consider multiple characteristics of product architecture from a single model. The HDDSM and accompanying analyses can be extended to products of varying complexity and scale.

5. Conclusion

A HDDSM has been introduced for modelling product architecture. The HDDSM builds upon existing DSM research in two ways that are beneficial for evaluating product architecture. First, the HDDSM is built with a modular modelling method that allows a practitioner to assemble a highly detailed system-level HDDSM from HDDSMs of subsystems, assemblies, or modules. The modular modelling method also facilitates distribution of the task of building an HDDSM among multiple practitioners, and it reduces the effort required to build a highly detailed HDDSM (Tilstra *et al.* 2009, Tilstra 2010).

Second, the HDDSM relies upon an interaction basis that defines a standardised set of interactions that can occur between elements in the HDDSM. The standardised interaction basis is intended to encompass the types of interactions that are important for analysing the architecture of electro- and thermo-mechanical products. The standardised interaction basis also allows HDDSMs for different products to be compared consistently, even if those HDDSMs are created by different examiners. Preliminary studies have shown sufficient levels of repeatability in HDDSMs created by different practitioners (Tilstra 2010, Tilstra *et al.* 2010). Also, in task-based DSMs, it has been noted that 'the binary matrix is often crowded with weak dependencies, and this leads to an extremely coupled design matrix' (Eppinger *et al.* 1994). By creating a standardised interaction basis, different types of element dependencies can be distinguished while requiring only a judgment of existence by the practitioner.

Another benefit of the interaction basis is an enhanced ability to analyse product architecture. For example, if electrical energy and mechanical energy are both being used within a product, modelling them both as energy interactions or, even more generally, as binary assessments of interaction, abstracts away from details of the product architecture that could be important for subsequent analysis. For example, Stone *et al.* (2000) describe how tracing these specific types of flows can lead to module identification.

Finally, the HDDSM provides a suitable foundation for quantitatively analysing various aspects of product architecture and comparing those analyses across products. This capability was illustrated in Section 4 by analysing a variety of consumer products and identifying those with frameworks and space potential for supporting evolutionary design changes. Future work could include expanding the types of architectural analysis that can be performed on the HDDSM and applying these analyses at various levels of detail, from component-level to system-level Element Detail, with the potential of strategically clustering elements as part of the analysis. This type of HDDSM-based analysis could be useful for not only comparing products within or across product lines, but also selecting and refining preliminary concepts during the design process. Multiple metrics would support multi-objective selection and optimisation of product architecture. It would also be interesting to further investigate the repeatability of the HDDSM, and to apply it to a broader variety of products. Furthermore, as the HDDSM becomes more established as a design tool, it could be valuable to integrate it into product lifecycle management systems that include BOM, computer-aided design (CAD) files, and other information that could be linked with the contents of the HDDSM.

Acknowledgements

The authors acknowledge the support provided from The University of Texas at Austin, the Cullen Trust for Higher Education, and the National Science Foundation under Grant No. CMMI-0600474. Any opinions, findings and conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

Notes

1. In addition to the BOM, CAD models can be used to inform the HDDSM model. CAD models are not required and may not always be available; however, if they are available, they could be used to identify some of the interactions listed in Table 1.
2. In its current form, it does not distinguish between intra- and inter-module interactions, as Sosa's approach does, but it could be expanded to do that by operating on HDDSMs of different levels of fidelity.

References

Albers, A., *et al.*, 2011. System architecture modeling in a software tool based on the Contact and Channel Approach (C&C-A). *Journal of Mechanical Design*, 133 (10), p. 101006 (8 pages).

Alizon, F., *et al.*, 2007. Three dimensional design structure matrix with cross-module and cross-interface analyses. *ASME IDETC Design Automation Conference*, Las Vegas, NV. Paper Number: DETC2007-34510.

Andreasen, M.M., 2011. 45 years with design methodology. *Journal of Engineering Design*, 22 (5), 293–332.

Boothroyd, G., Dewhurst, P., and Knight, W., 2002. *Product design for manufacture and assembly*. 2nd ed. New York, NY: Marcel Dekker.

Browning, T.R., 2001. Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Transactions on Engineering Management*, 48 (3), 292–306.

Clarkson, P.J., Simons, C., and Eckert, C., 2004. Predicting change propagation in complex design. *Journal of Mechanical Design*, 126 (5), 788–797.

Eckert, C., Clarkson, P., and Zanker, W., 2004. Change and customisation in complex engineering domains. *Research in Engineering Design*, 15 (1), 1–21.

Eppinger, S.D., *et al.*, 1994. A model-based method for organizing tasks in product development. *Research in Engineering Design*, 6 (1), 1–13.

Fricke, E. and Shulz, A.P., 2005. Design for changeability (dfc): principles to enable changes in systems throughout their entire lifecycle. *Systems Engineering*, 8 (4), 342–358.

Helmer, R., Yassine, A., and Meier, C., 2010. Systematic module and interface definition using component design structure matrix. *Journal of Engineering Design*, 21 (6), 647–675.

Hirtz, J., *et al.*, 2002. A functional basis for engineering design: reconciling and evolving previous efforts. *Research in Engineering Design*, 13 (2), 65–82.

Hölttä, K., Suh, E.S., and de Weck, O., 2005. Tradeoff between modularity and performance for engineered systems and products. *International Conference on Engineering Design*, Melbourne, Australia.

Hsiao, S.W. and Liu, E., 2005. A structural component-based approach for designing product family. *Computers in Industry*, 56 (1), 13–28.

Jiao, J., Simpson, T.W., and Siddique, Z., 2007. Product family design and platform-based product development: a state-of-the-art review. *Journal of Intelligent Manufacturing*, 18 (1), 5–29.

Keese, D., *et al.*, 2007. Empirically-derived principles for designing products with flexibility for future evolution. *ASME IDETC Design Theory and Methodology Conference*, Las Vegas, NV. Paper Number: DETC2007-35695.

Kuchinsky, B., 2005. *Development of product flexibility principles for engineering design*. Thesis (MS). Mechanical Engineering, University of Texas, Austin, TX.

Luh, D.-B., Ko, Y.-T., and Ma, C.-H., 2011. A structural matrix-based modelling for designing product variety. *Journal of Engineering Design*, 22 (1), 1–29.

Martin, M.V. and Ishii, K., 2002. Design for variety: developing standardized and modularized product platform architectures. *Research in Engineering Design*, 13 (4), 213–235.

Maurer, M. and Lindemann, U., 2008. The application of the Multiple-Domain Matrix. *IEEE International Conference on Systems, Man and Cybernetics*, Singapore, 2487–2493.

Otto, K.N. and Holtta-Otto, K., 2007. A multi-criteria assessment tool for screening preliminary product platform concepts. *Journal of Intelligent Manufacturing*, 18 (1), 59–75.

Otto, K.N. and Wood, K.L., 2001. *Product design*. Upper Saddle River, NJ: Prentice-Hall.

Pimmler, T.U. and Eppinger, S.D., 1994. Integration analysis of product decompositions. *ASME DETC Design Theory and Methodology Conference*, Minneapolis, MN, 343–351.

Qureshi, A.M., *et al.*, 2006. Principles of product flexibility. *ASME IDETC Design Theory and Methodology Conference*, Philadelphia, PA. Paper Number: DETC2006-99583.

Rothwell, R. and Gardiner, P., 1988. Re-innovation and robust designs: producer and user benefits. *Journal of Marketing Management*, 3 (3), 372–387.

Simpson, T.W., 2004. Product platform design and customization: status and promise. *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing*, 18 (1), 3–20.

Sosa, M.E., Eppinger, S.D., and Rowles, C.M., 2003. Identifying modular and integrative systems and their impact on design team interactions. *Journal of Mechanical Design*, 125 (2) (June), 240–252.

Steward, D.V., 1981. The design structure system: a method for managing the design of complex systems. *IEEE Transactions on Engineering Management*, 28 (3), 71–74.

Stone, R., Wood, K.L., and Crawford, R.H., 2000. A heuristic method for identifying modules for product architecture. *Design Studies*, 21 (1), 5–31.

Suh, E., Weck, O., and Chang, D., 2007. Flexible product platforms: framework and case study. *Research in Engineering Design*, 18 (2), 67–89.

Tilstra, A., 2010. *Representing product architecture and analyzing evolvable design characteristics*. Dissertation (PhD). Mechanical Engineering, The University of Texas at Austin, Austin, TX.

Tilstra, A.H., Seepersad, C.C., and Wood, K.L., 2009. Distributed modeling of component DSM. *International Design Structure Matrix Conference*, Greenville, SC.

Tilstra, A., Seepersad, C.C., and Wood, K.L., 2010. The repeatability of high definition design structure matrix (HDDSM) models for representing product architecture. *ASME IDETC Design Theory and Methodology Conference*, Montreal, Quebec. Paper Number: DETC2010-28717.

Ulrich, K., 1995. The role of product architecture in the manufacturing firm. *Research Policy*, 24 (3), 419–440.

Warfield, J.N., 1973. Binary matrices in system modeling. *IEEE Transactions on Systems, Man, and Cybernetics SMC*, 3 (5), 441–449.