10TH INTERNATIONAL DESIGN STRUCTURE MATRIX CONFERENCE, DSM'08
11 – 12 NOVEMBER 2008, STOCKHOLM, SWEDEN

ANALYSIS OF HYBRID VEHICLE ARCHITECTURES USING MULTIPLE DOMAIN MATRICES

Carlos Gorbea[1, 2], Tobias Spielmannleitner[1], Udo Lindemann[1] and Ernst Fricke[2]
[1]Technische Universität München, Institute for Product Development
[2] BMW Group

Keywords: Multiple Domain Matrix, Vehicle Architecture, Hybrid Electric Vehicles

1 INTRODUCTION

This work presents a methodology to analyze various hybrid electric vehicle (HEV) architectures by means of Multiple Domain Matrices (MDM), with the goal of exploring how functions and components relate for any chosen HEV configuration. The analysis of dependencies between and amongst the *functional* and *component* domains of eight different HEV architectures developed a basis of knowledge to define a total of 76 sensible possible architectures out of a field of more than 432 different combinations. **ΔMDMs** and **∑MDMs** where used in comparing individual HEV architectures and enabled the development of a rule based synthesis of HEV architectures.

2 MOTIVATION FOR HEV ARCHITECTURE ANALISYS

Knowing that cars are complex systems built on numerous sub-systems, **vehicle architecture** then refers to the linking of vehicle sub-systems in a particular configuration to meet a desired set of functions [1]. As automakers commence to enter a new age of architectural competition, there exists a need for analysis tools that develop a clear understanding of which future car configurations will best deliver the firm's requirements [2]. The MDM representation of dependencies between components and their basic functions is used as a tool to analyze these complex structures in an organized manner. In this paper, we specifically limit ourselves to combinations of electric motors and internal combustion engine configurations as a first step before expanding the field to alternative fuel converters such as fuel cells, gas turbines, etc.

3 ANALYSIS OF ARCHITECTURES USING ΔMDM

Figure 1 depicts how ΔMDMs can be used in comparing two HEV architectures. First, MDMs for the car architectures considered are built. These are constructed by first building a 'component DSM' that depicts physical component sub-system dependencies in a symmetrical matrix (bottom right matrix); a 'functions-components DMM' (bottom left matrix) showing dependencies between functions and components; and finally a 'functions DSM' (top right matrix) that shows dependencies within functions. Once any two vehicle architecture MDMs are available, subtracting one MDM to another creates a so called ΔMDM that is useful in comparing differences amongst the two architectures. This process is analogous to similar work considering ΔDSMs [5]. By means of a ΔMDM, engineers can readily compare how differences between vehicle components affect the functional domain.

The differences captured by a ΔMDM also serve an important role in checking whether the initial matrices where filled correctly by identifying the limited number of matrix cells that differ between the two. A further expansion of the MDMs to include a "requirements" and a "cost" domain in a 'four-domain' ΔMDM, could show how differences in components affect cost and the fulfilment of requirements.

An important point to consider before calculating ΔMDMs is that the functional and component domain indices for both architectures must be the same and the MDM fields must contain binary logic (cells contain either 1 or 0). Hence, all components and all functions relevant to both architectures considered must have identical fields in the MDMs to facilitate the calculation of the ΔMDM.

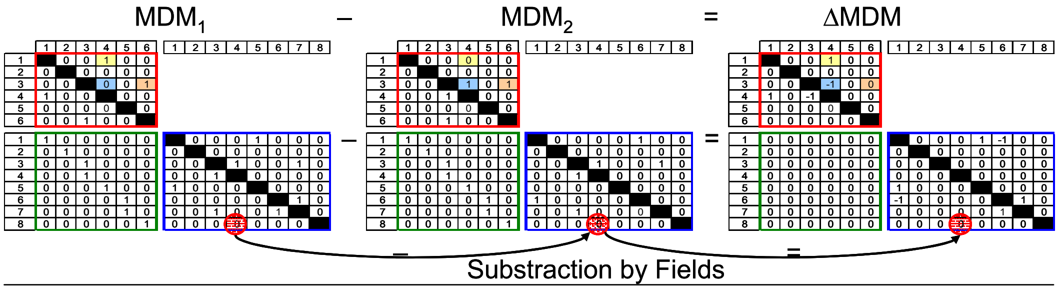


Figure 1: Given two HEV Architectures where MDMs have been documented, a ΔMDM shows point differences between architectures

4 ANALYSIS OF ARCHITECTURES USING ∑MDM

The ∑MDM is a summation of two or more MDMs pertaining to vehicle architectures as depicted in figure 2. Similar to the ΔMDM, the matrices being added must match in terms of the function and component elements and indexes. This methodology is analogous to similar DSM work presented by several authors [6, 7].

In this study we originally built eight MDMs pertaining to different car architectures [internal combustion engine, parallel (micro) hybrid – belt assisted starter/generator, parallel (mild) hybrid – integrated motor assist, parallel hybrid – double coupling, power split hybrid - one mode, power split hybrid – two mode, series hybrid, electric vehicle]. The addition of these eight MDMs enabled the determination of which components apply to all architectures (cells showing a sum of 8) and which components where found to vary across architectures.

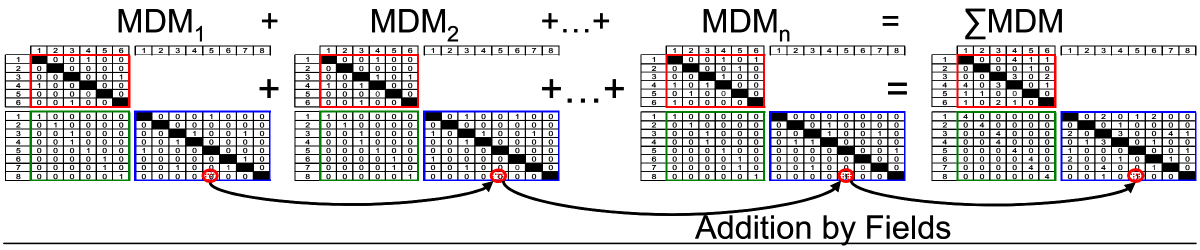


Figure 2: Given n HEV architecture MDMs, the ∑MDM reveals patterns of which dependencies are always present between and amongst the component and function domains

A generic DMM for components across all eight architectures was build using the information from the ∑MDM. This resulted in a valuable matrix that allows for the calculation of either the functions DSM or components DSM given that either one can be built [3]. For example, equation 1 shows how a functions DSM can be derived from a components DSM according to Maurer.

$$\text{Functions DSM} = (\text{F-C DMM})^T \times (\text{Components DSM}) \times (\text{F-C DMM}) \qquad (1)$$

5 GENERATING RULES TO BUILD HEV COMPONENT DSM

Learning from the structural analysis knowledge gained from the eight MDMs, the ΔMDMs, the ∑MDM and research on HEV configurations [4], we were able to develop the typology fields for HEV architectures shown in figure 3. We purposely include the conventional ICE and the electric vehicle (EV) architectures to complete the entire ICE-HEV-EV spectrum.

Architecture Type	Drive Configuration	Transmission Type	E-Motor Placement
Conventional ICE	2WD	Automatic	Pre-Transmission
Parallel Hybrid	4WD	Manual	Post-Transmission
Series Hybrid	2+2WD	CVT	Belt Alternator
Power Split		No Transmission	Wheel Motor
Series-Parallel		Planet Gears	
Electric Vehicle		Torque Coupling	

Figure 3: Simple typology of ICE-HEV-EV architectures

The combination of these fields results in a total of 432 different vehicle configurations. Using the results of the ∑MDM generated from the initial architectures, a set of synthesis rules were collected to determine that only 76 of these architectures were sensible. For example, some components were found to be always present (ie. the battery, the cooling system, and the wheels to name a few), while others were found to show defined dependencies. Such observations led to a rule based generation of 'components DSM' for any one of the 76 typologies.

6 DESIGN SYNTHESIS OF HEV ARCHITECTURES

With the rule based creation of the components DSM, and the establishment of a generic DMM derived from the ∑MDM analysis, the dependencies of the functional domain can be calculated using equation 1. Because the 'components DSM' is a symmetrical matrix, the calculated 'functions DSM' also results in a symmetrical matrix. However, this calculated 'functions DSM' can be easily converted to a non-symmetrical matrix by considering energy and information flow rules. The creation of a 'functions DSM' that follows an input-output logic has been shown to allow for clustering of basic functions into higher order functions [1].

7 SUMMARY

The use of MDMs in analyzing HEV architectures was presented. Once several baseline architectures can be described in an MDM format, the utility of performing basic subtraction and addition of architectures by means of ΔMDM and ∑MDM provide valuable information that can lead to an exploration of many more possible solutions. Ongoing research lies in using ΔMDM and ∑MDM information for an automated generation of a solution space using design synthesis rule based methods. The following list of steps summarizes the procedure being investigated:

1. Several known car architectures structures are constructed in MDM format
2. Individual architecture MDMs are compared using subtraction to form a ΔMDM analysis
3. A ∑MDM of all known architectures reveals patterns for rule generation, HEV typology, and a generic DMM for across all architectures is assembled
4. A 'components DSM' is generated from the rules in step 3
5. A 'functions DSM' is calculated using equation 1 to complete the MDM structure
6. The calculated 'functions DSM' is revised to fit a input-output logic based on energy and information flow rules
7. Clustering of 'functions DSM' reveals customer relevant functions

REFERENCES

[1] Gorbea C., Fricke E., Lindemann U., "Pre-Selection of Hybrid Electric Vehicle Architectures during the Initial Design Phases", Procedings of the 9th International DSM Conference 2007.

[2] Gorbea C., Fricke E., Lindemann U., "The Design of Future Cars in a New Age of Architectural Competition", Proceedings of the ASME IDETC/CIE '08, August 2008.

[3] Maurer M., *Structural Awareness in Complex Product Design*, Dissertation Technischen Universitaet Muenchen, Verlag Dr. Hut, October 2007.

[4] Rousseau A. et. al, Power Train Systems Analysis Toolkit User Documentation, Aragonne National Laboratories, 2008.

[5] Smaling R., de Weck O., "Assesing Risks and Opportunities of Technology Infusion in System Design", *Systems Engineering*, 10 (1): 2007.

[6] Braun T. and Deubzer F., "New Variant Management Using Multiple-Domain Mapping", Proceedings of the 9th International Design Structure Matrix Conference 2007.

[7] Gausemeier J., et al., "Using DSM for the Modularization of Self-Optimizing Systems", Proceedings of the 9th International Design Structure Matrix Conference 2007.

Contact: Carlos Gorbea
BMW Group / Technische Universität München, Institute for Product Development
Boltzmannstraße 15
85748 Garching, Germany
+49.89.382.78718
+49.89.289.15144
carlos.gorbea@bmw.de

10TH INTERNATIONAL DSM CONFERENCE

Analysis of Hybrid Vehicle Architectures using Multiple Domain Matrices

Ing. Carlos Gorbea, MBA, PE*
(Technische Universität München-Institute for Product Development / BMW Group)

Dipl. Ing. Tobias Spielmannleitner
(Technische Universität München-Institute for Product Development)

Prof. Dr. Ing Udo Lindemann
(Technische Universität München-Institute for Product Development)

Dr. Ing. Ernst Fricke
(BMW Group)

***Corresponding Author**

Technische Universität München

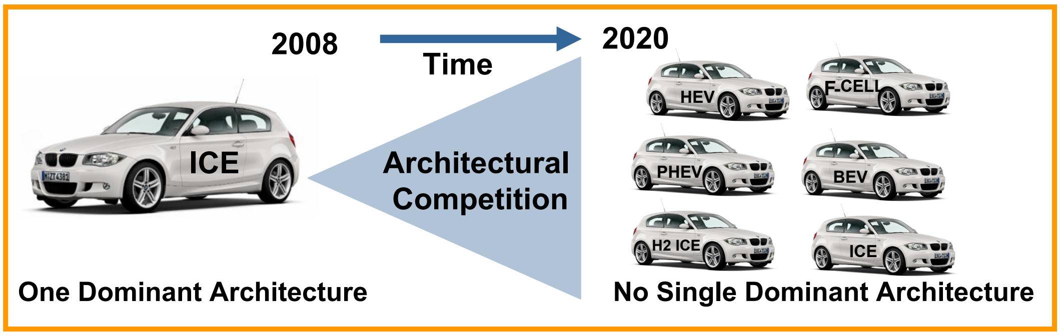
MANAGE COMPLEX SYSTEMS
FOLLOW THE FLOW OF INFORMATION!

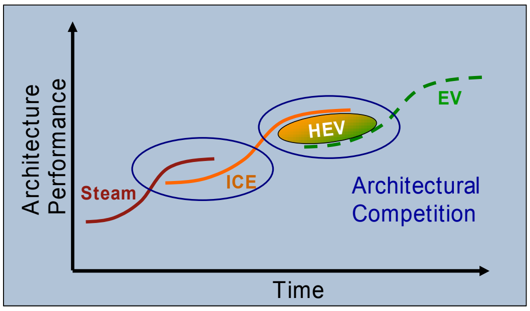
Agenda

- Motivation and Problem Description
- MDM Analysis Methodology
- ΔMDMs and ∑MDMs for Architecture Analysis
- Design Synthesis and Summary

Technische Universität München

We have entered a new age of "Architectural Competition"





How many power train architectures are there in the Electrification of Cars?

-A methodical approach is needed to help vehicle architects understand the relationships between form and function.

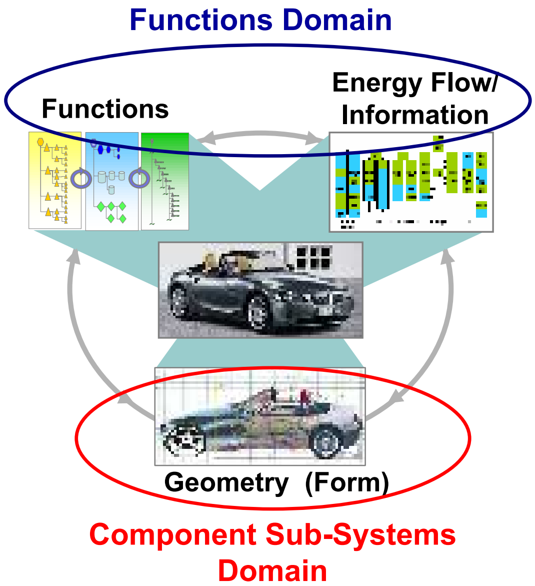
Source: Gorbea, Fricke, Lindemann (2008)

FOLLOW THE FLOW OF INFORMATION!
MANAGE COMPLEX SYSTEMS

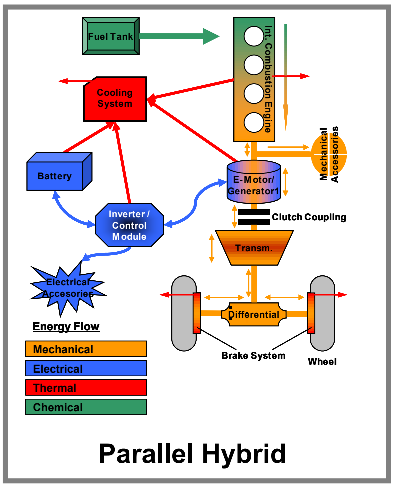
What is meant with "Vehicle Architecture"?

Vehicle Architecture refers to the linking of vehicle sub-systems in a particular configuration (form) to meet a desired set of functions, that allow the required Information and energy flow to occur.

•In this study we focus specifically on the power train architectures that are simplified using abstraction

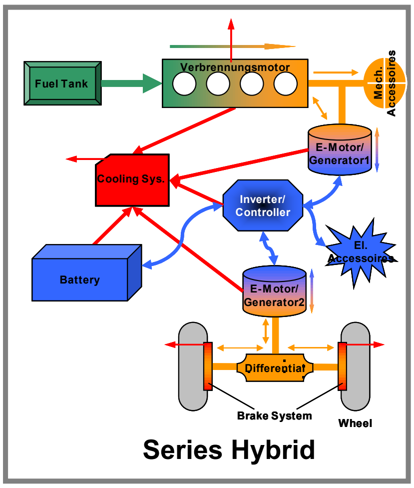


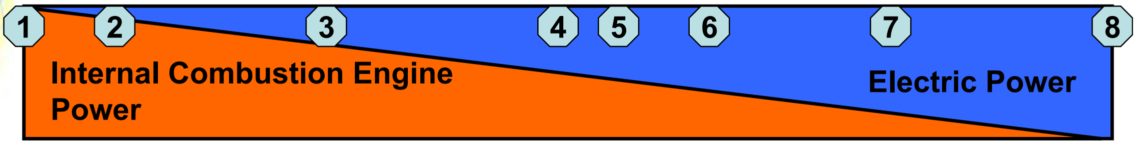
There exists a varied spectrum of vehicle architectures



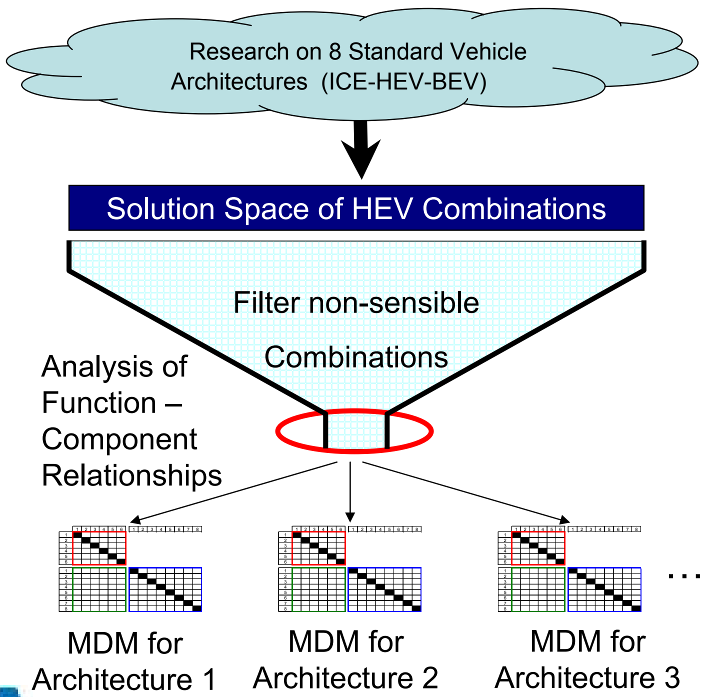
8 base architectures were selected for initial analysis:

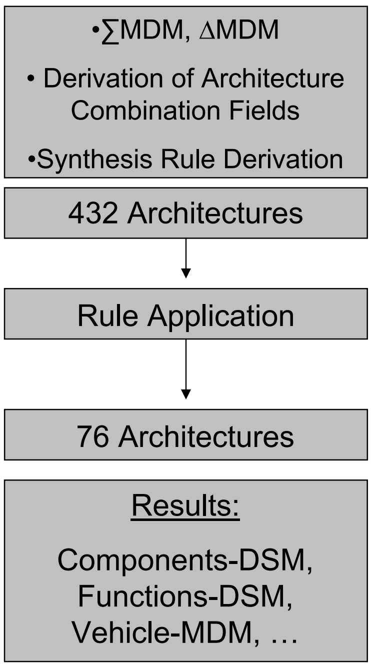
1. Standard ICE
2. Parallel (BAS)
3. Parallel (ISG)
4. Parallel (P2)
5. Split (1 Mode)
6. Split (2 Mode)
7. Series Hybrid
8. Battery Electric



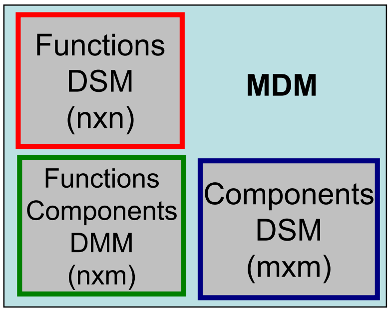


Methodical overview of the vehicle architecture analysis to follow





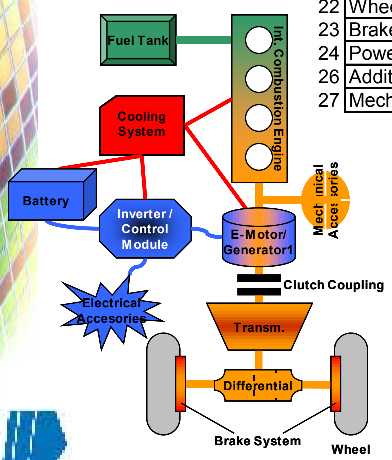
Describing vehicle architectures with Multiple Domain Matrices



		Store Fuel	Store Electric Energy	Convert Fuel into Mechanical Energy	Convert Mechanical into Electrical Energy	Convert Electrical into Mechanical Energy	…	Fuel Tank	High Voltage Battery	Internal Combustion Engine	Generator	Starter	…
		1	2	3	4	5	…	20	21	22	23	24	…
Store Fuel	1			1									
Store Electric Energy	2					1							
Convert Fuel into Mechanical Energy	3				1								
Convert Mechanical into Electrical Energy	4		1			1							
Convert Electrical into Mechanical Energy	5				1								
⋮	6												
Fuel Tank	20	1								1			
High Voltage Battery	21		1								1	1	
Internal Combustion Engine	22			1				1			1	1	
Generator	23				1				1	1			
Starter	24					1			1	1			
⋮	⋮												

The ‘Components DSM’ shows physical connections

		1	2	4	5	11	13	18	21	22	23	24	26	27
1	Fuel Tank			1										
2	High Voltage Battery								1			1		
4	Internal Combustion Engine	1			1				1					1
5	E-Motor/Generator1			1				1	1			1		
11	Transmission						1	1						
13	Differential Gear					1				1				
18	Clutch Direct Coupling1				1	1								
21	Cooling System		1	1	1							1		
22	Wheels						1				1			
23	Brake-system									1				
24	Power Electronics/Inverter		1		1				1				1	
26	Additional Electric Accessories											1		
27	Mechanical Accessories			1										



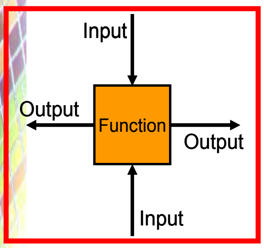
The Components DSM is a Symmetric Matrix

- Describes physical component connections
- Example: ‘Fuel tank’ – ‘Combustion Engine'
- A graphic representation of the matrix can be depicted

The 'Functions DSM' depicts information and energy flow dependencies

Uni-Directional Relationship

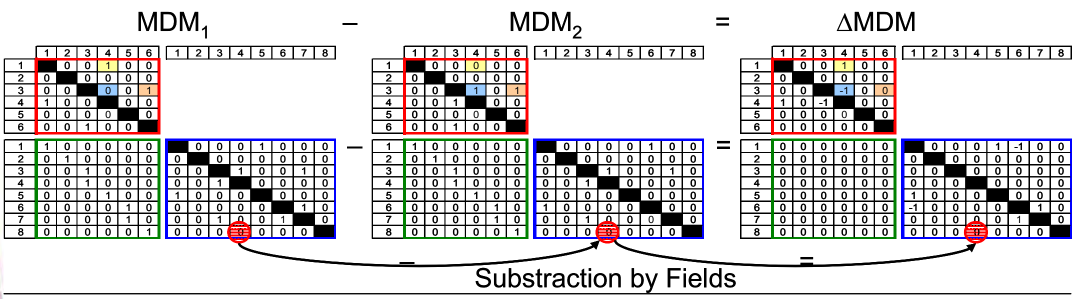
Bi-Directional Relationship



Functions DSM	Store Fuel	Store Electric Energy	Convert Fuel in Mechanical Energy	Convert Mechanical into Electrical Energy	Convert Electrical into Mecanical Energy	...
Store Fuel			1			
Store Electric Energy					1	
Convert Fuel in Mechanical Energy			Convert Fuel in Mechanical Energy	1		
Convert Mechanical into Electrical Energy		1			1	
Convert Electrical into Mecanical Energy				1		
⋮						

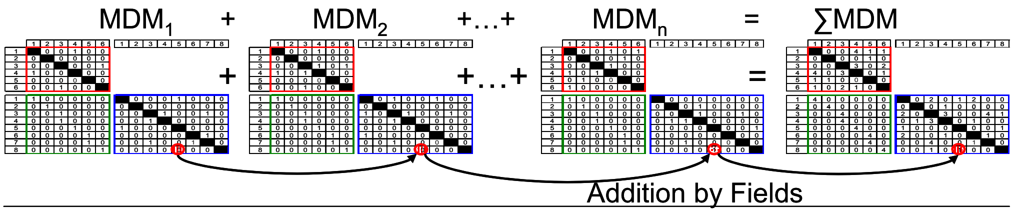
The Functions DSM is a Non-Symmetric Matrix

Comparison of two vehicle architectures by means of a ΔMDM



- Simple matrix subtraction – analogous to ΔDSMs (De Weck 2007)
- Allows for easy detection of differences between architectures
- Fields being compared must be consistent with each other
- Can be used as a check to see if mistakes have been made in filling out matrices

∑MDM allow for further analysis of vehicle architectures



- Analogous to DSM matrix addition – (Braun and Deubzer 2007; Gausemeier 2007 and others)
- Allows for deriving synthesis rules applicable to all architectures
  - ➢Example: "If 'fuel tank' is a Component, then an 'engine' must be present"
- Allows to gather system information on which components serve more than one function ('data mining' to find relationships between elements in two domains)

➢ A ∑MDM in our methodology results in the construction of a generic DMM is found that applies to all vehicle architectures enabling for calculation of either the 'Functions DSM' or the 'Component DSM'

A ∆MDM and ∑MDM analysis of the 8 base architectures revealed four key structural elements that differentiate architectures

•Elements that determine vehicle structure

1 Architecture Type	2 Drive Configuration	3 Transmission Type	4 E-Motor Placement
Conventional ICE	2WD	Automatic	Pre-Transmission
Parallel Hybrid	4WD	Manual	Post-Transmission
Series Hybrid	2+2WD	CVT	Belt Alternator
Power Split		No Transmission	Wheel Motor
Series-Parallel		Planet Gears	
Electric Vehicle		Torque Coupling	

•Structural Tree (follows from the combinations of the elements above)

Electric Vehicle	2WD	Torque Coupling	Wheel E-Motor Before Transmission
		No Transmision	Wheel E-Motor
	4WD	Torque Coupling	Wheel E-Motor Before Transmission
		No Transmision	Wheel E-Motor
Conventional	2WD	Automatic Transmission	Starter/ Generator (Alternator)
		CVT	Starter/ Generator (Alternator)
⋮	⋮	⋮	⋮

432 Combinations→ 76 Valid Combinations

∆MDM and ∑MDM analysis allowed for a collection of synthesis rules

The following components always exhibit physical connections in the components DSM:

- Battery – Starter
- Battery – Generator
- Battery – Cooling System
- Battery – Inverter/Controller
- CVT – Automatic Torque Converter
- E-Motor – Inverter/Controller
- E-Motor1/2/3 – Cooling System
- Cooling System – Inverter/ Controller
- Inverter/ Controller – El. Accessories
- Inverter/Controller – Plug
- Planet Gear – E-Motor
- Wheel – Brake system

⋮

- Wheel – Differential
- Wheel E- Motor – Inverter/Controller
- Wheel E- Motor – Wheel
- Super Capacitor – Inverter/Controller
- Combustion Engine– Starter
- Combustion Engine– Mech. Accessories
- Combustion Engine– Generator
- Combustion Engine– Fuel Tank
- Combustion Engine– Cooling System
- Combustion Engine– Planet Gear
- Comb. Engine– Auto. Torque Converter
- Transfer Case Gearbox– Differential

⋮

∆MDM and ∑MDM analysis allowed for a collection of synthesis rules

Category 1:
The following components are found in all Architectures:
- Battery, Cooling System, Wheel, Brake System, Electrical Accessories

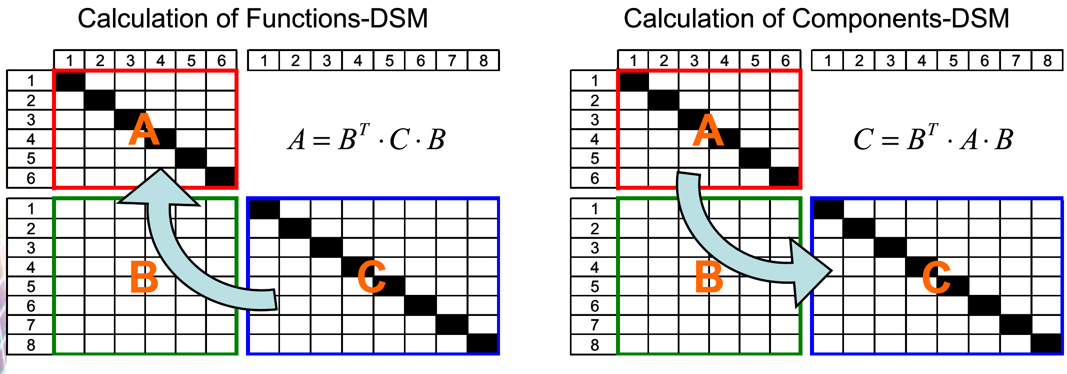
Category 2:
Rules for specific components:
- Transfer Case Gearbox always with 4WD, except where Wheel E- Motor is present (no transmission)
- If combustion engine is present then always a tank and mechanical accessories will be present
- Automatic transmission always with Automatic Torque Converter
- Clutch Coupling with Manual Transmission
- Power split architectures always have Planet Gear and E-Motors

Category 3:
Rules for particular architectures:
- Except for architectures with wheel motors, a differential is present
- Conventional and BAS architectures: Generator und Starter present
- Planet Gear only as transmission in Power split architectures
- 2+2WD only in Parallel hybrids
- Combustion Engine in all architectures except electric vehicle
- All Architectures, except conventional have a Inverter/Electronic Controller
- A torque coupling is present when E-Motor before the differential
- Plug only in plug-in-hybrid and electric vehicles
- Automatic Torque Converter can replace a direct coupling except in a double coupling transmission

FOLLOW THE FLOW OF INFORMATION!

MANAGE COMPLEX SYSTEMS

A generic DMM across architectures allows for calculation of DSMs



•An unknown DSM in an MDM can be calculated from a related DMM and DSM as shown above (Maurer 2007)

• Using this principle, once a generic DMM is derived for all vehicle architectures, it can be used to calculate functional dependencies from a given set of component dependencies or vice-versa

Technische Universität München

JÖNKÖPING INTERNATIONAL BUSINESS SCHOOL

FOLLOW THE FLOW OF INFORMATION!

MANAGE COMPLEX SYSTEMS

Applying 'Input-Output' logic to a calculated 'Functions-DSM'

A generic functions DSM template converts the calculated symmetric Functions-DSM to one that contains the directional flow of energy and information

	Store Fuel	Store Electric Energy	Convert Fuel into Mechanical Energ	Convert Mechanical into Electrical E	Convert Electrical into Mechanical E	Deliver (Recover) torque to (from) v	Convert Moment tranfered (mechar	Equate Rotation	Divide Moment between Wheel Co	Couple/Uncouple Moment	Release Energy as Heat to the Env	Transfer Heat (to Cooling system)	Transfer Moment to (from) the road	Slow or Stop Vehicle (recovering e	Slow or Stop Vehicle (releasing En	Control Energy Flow	Connect/Transfer External Electric	Consume Electric Energy for Auto /	Consume Mechanical Energy for E
Store Fuel			1									1							
Store Electric Energy											1	1				1			
Convert Fuel into Mechanical Energy	1			1	1						1	1		1					1
Convert Mechanical into Electrical Energy			1							1	1	1				1			
Convert Electrical into Mechanical Energy			1							1	1	1				1			
Deliver (Recover) torque to (from) wheels							1						1	1	1				
Convert Moment tranfered (mechanical)						1		1		1			1	1	1				
Equate Rotation							1						1	1	1				
Divide Moment between Wheel Connections																			
Couple/Uncouple Moment				1	1		1					1		1					
Release Energy as Heat to the Environment		1	1	1	1							1	1	1	1	1			
Transfer Heat (to Cooling system)	1	1	1	1	1					1	1			1		1		1	1
Transfer Moment to (from) the road						1	1	1			1				1				
Slow or Stop Vehicle (recovering energy)			1			1	1	1		1	1	1			1	1			
Slow or Stop Vehicle (releasing Energy - by Friction)						1	1	1			1		1	1					
Control Energy Flow		1		1	1						1	1		1				1	
Connect/Transfer External Electric Energy Source																			
Consume Electric Energy for Auto Accesory OPS												1				1			
Consume Mechanical Energy for Engine Accesory			1									1							

•Only green fields are allowed, gray fields removed

Technische Universität München

JÖNKÖPING INTERNATIONAL BUSINESS SCHOOL

Synthesis of new HEV architectures

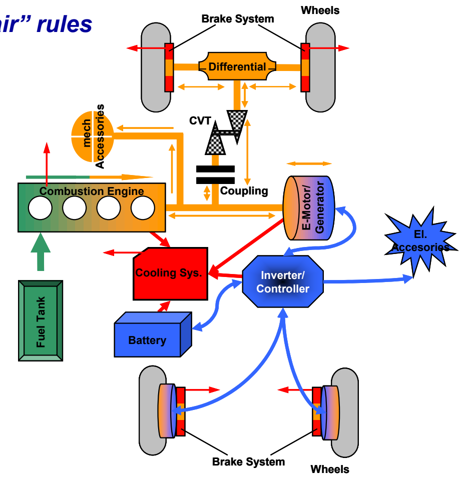
Step 1: Select 1 of 76 available vehicle architecture combinations

Type	Drive	Transmission	E-Motor Placement
Parallel Hybrid	2+2WD	CVT	Wheel Motors

Step 2: Apply "architecture" and "component pair" rules

<u>Rules for Component Pairs</u>

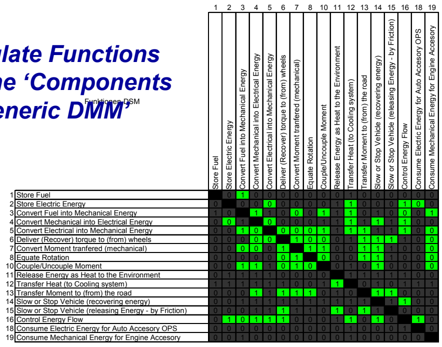
- Battery – Cooling System
- Battery – Inverter/Controller
- E-Motor1/2/3 – Inverter/Controller
- E-Motor1/2/3 – Cooling System
- Cooling System – Inverter/Controller
- Inverter/Controller – El. Accessories
- Combustion Engine– M. Accessories
- Wheel – Brake system
- Wheel – Differential
- Wheel E- Motor – Wheel
- CVT – Clutch Coupling
- Wheel E- Motor – Inverter/Controller
- Combustion Engine– Fuel Tank
- Combustion Engine– Cooling System

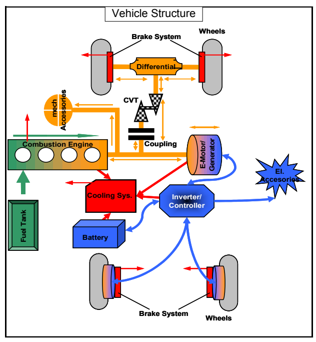


Step 3: Build the Structure / Components DSM

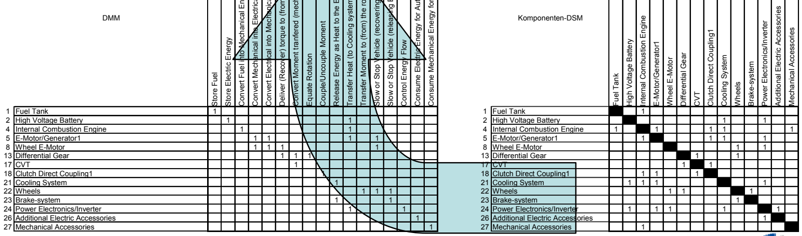
MDM for "Through the Road Hybrid with Wheel Motors"

Step 4: Calculate Functions DSM using the 'Components DSM' and 'Generic DMM'





Architecture Analysis Functions-Component



Clustering the ‘Functions DSM’ reveals ‘Customer Functions’

Step 5: Cluster Analysis reveals higher order functions

Functions DSM

Functions DSM	Convert Mechanical into Electrical Energy	Slow or Stop Vehicle (recovering energy)	Control Energy Flow	Store Electric Energy	Convert Electrical into Mechanical Energy	Transfer Moment to (from) the road	Deliver (Recover) torque to (from) wheels	Moment wandeln (mechanisch)	Convert Moment tranfered (mechanical)	Store Fuel	Convert Fuel into Mechanical Energy	Equate Rotation	Slow or Stop Vehicle (releasing Energy - by Friction)	Transfer Heat (to Cooling system)	Consume Electric Energy for Auto Accesory OPS	Consume Mechanical Energy for Engine Accesory	Release Energy as Heat to the Environment
Convert Mechanical into Electrical Energy	■	1	1											1			
Slow or Stop Vehicle (recovering energy)		■	1														
Control Energy Flow	1	1	■	1	1			1						1	1		
Store Electric Energy			1	■										1			
Convert Electrical into Mechanical Energy			1		■	1			1		1			1			
Transfer Moment to (from) the road	1	1				■	1	1				1	1				
Deliver (Recover) torque to (from) wheels						1	■	1					1				
Moment wandeln (mechanisch)						1	1	■	1			1					
Convert Moment tranfered (mechanical)	1							1	■		1						
Store Fuel										■	1						
Convert Fuel into Mechanical Energy	1								1		■			1		1	
Equate Rotation						1		1				■					
Slow or Stop Vehicle (releasing Energy - by Friction)						1	1						■				1
Transfer Heat (to Cooling system)														■			1
Consume Electric Energy for Auto Accesory OPS															■		
Consume Mechanical Energy for Engine Accesory																■	
Release Energy as Heat to the Environment																	■

Regenerative Braking
Boosten
Drive Electric
Differential
Drive with Combustion Engine

Summary

- We have entered a new age of architectural competition in the automotive industry
- Vehicle architecture refers to the linking of vehicle sub-systems in a particular configuration to meet a desired set of functions
- Multiple Domain Matrices allow for an effective way to analyze vehicle architectures – showing links amongst functions and component domains
- A methodology was presented in which 76 vehicles architectures within the ICE-HEV-BEV spectrum can be analyzed from 8 base architectures
- The methodology presented can be expanded upon by further defining functions and components found in the various architectures (ie. less abstraction and more detail)
- Only when overall vehicle goals/requirements are defined can a true comparison amongst car architectures be achieved
  - Need three items to allow a valid comparison amongst two cars:
    1. Structure, 2. Dimensioning and 3. Control Strategy
  - The latter two points can only be determined if vehicle goals/requirements are available!