Current Trends in Network Development for Heavy-Duty Vehicles

Factors of success in electronic development

ECU networking in heavy-duty vehicles is characterized by the same challenges as in the automobile. Added difficulties are caused by the large numbers of variants with low production volumes and longer product life cycles, requiring a suitable architecture layout. Specially modified development methods are indispensable in handling cost pressure and sending reliable vehicles onto the street.

The number of ECUs, and hence the amount of software, has multiplied since electronification began in the early 1990s. While this primarily related to the engine controller at the beginning, a large number of electronic "helpers" are being implemented today. Examples include ABS, ESP, ACC and other driver assistance systems that make highway traffic safer and driving more pleasant. Analyses [1] assume that their implementation will increase further, and that electronic control modules will account for 90% of all innovations by the year 2010. A key aspect is that 80% of these innovations will exclusively involve software or the functions implemented in software. In this context, it is clear that software development methods play a crucial role in the development process for the total vehicle, and they have a significant influence on a vehicle's success or failure on the market.

Compared to automobiles, heavy-duty vehicle manufacturers are confronted by the special challenges of the relatively large number of variants with significantly lower volumes. Although simultaneous use of electronic ECUs over different brands and integration of standardized components can reduce cost pressure, they make the design of electronics and software more complex.

Flexible solutions are in demand

When one considers the variety of strategies used by different heavy-duty vehicle manufacturers, it quickly becomes clear that there is no universal solution. However, from a bird's eye perspective clear trends can be seen, such as the use of standards, code generators and a universal tool chain. The number of ECUs is increasing at a rather moderate rate, while the number of functions implemented purely in software continues to grow rapidly.

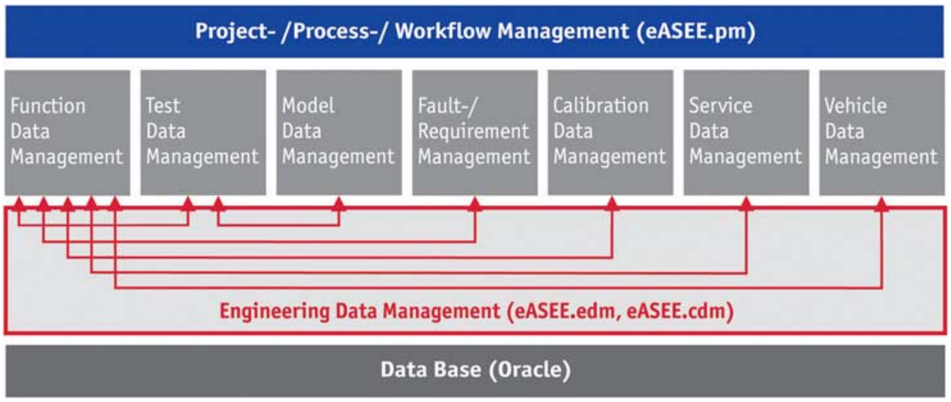
Common to solution strategies is the use of a comprehensive and universal tool chain – from requirements to validation. The use of individual, non-coordinated tools proved to be impractical in the past. The configuration processes and work results of individual tools are too different. This makes it difficult to achieve universality of change requirements during development. Thus, a change would need to be made in different tool configurations without any automatic, inter-tool consistency check. This causes organizational friction losses, especially in inter-departmental or inter-site development projects.

Therefore, a database with authoring tools should stand at the center of the tool chain. Both the database and authoring tools need to be specifically adapted to the requirements of the specific vehicle manufacturer. Besides purely technical aspects, the tools also take the individual development process of the companies into account. Variants management, configuration management and even the maintenance of workflows are represented in the tools. If external suppliers need to be integrated in processes, the data exchange formats that are used may be standards or de-facto industry standards such as the CANdb++ data format by Vector Informatik. In some cases, the vehicle manufacturer also prescribes the use of certain tools to its suppliers. They are then tightly coupled to the database and support the supplier especially in such aspects as compatibility to requirements, quality and efficiency. Examples would be code generators for embedded systems or test tools such as the CANoe.J1939 development and test tool from Vector.

System design is becoming increasingly complex due to growing requirements for networking. Individual ECUs are being installed on different platforms and different countries, which increases the number of variants. This requires flexibility in communication structures and in mapping functions to ECUs. This not only affects the available signals, but also the use of communication protocols. In Europe, for example, proprietary communication protocols are often used, which is similar to the situation in the automotive industry there. In the North American market, however, the SAE J1939 protocol dominates for heavy trucks. There are also differences in the area of in-vehicle diagnostics: In Europe, OBD diagnostics is implemented per UDS (ISO15765), and in the USA per SAE J1939-73.

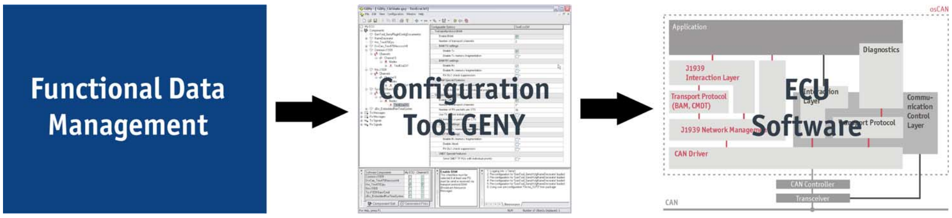
Attaining the goal by different approaches

The approach at MAN Nutzfahrzeuge AG is based on use of an integrated development database known as the “Common Engineering Data Backbone”. All vehicle-specific functions are developed from this platform, and all vehicle-specific information is stored there. The eASEE Tool Suite from Vector – with its 8 domains – serves as a universal development database, and it was specially adapted to requirements at MAN in the framework of a configuration process (Figure 1). It serves the needs of functional development and as a description of the communication matrix. Since MAN relies on the SAE J1939 standard as much as possible in communication, eASEE was adapted to the requirements of the J1939 protocol.



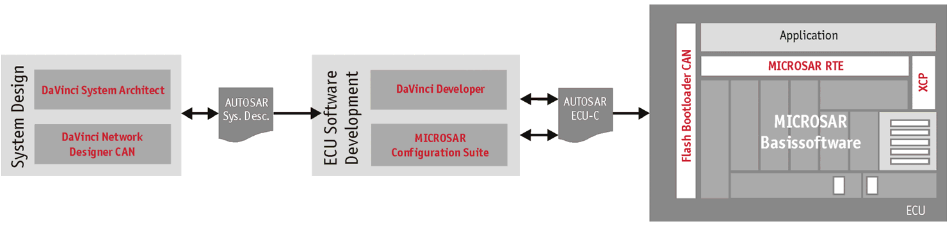
[Figure 1: The MAN Common Engineering Data Backbone]

A special module that was developed for MAN and adapted to the Data Backbone serves as a bridge between modeling in eASEE and automatic code generation for the ECUs (Figure 2). In code generation, the Munich-based heavy-duty vehicle producer relies on proven CANbedded.J1939 standard software components from Vector. CANbedded.J1939 gets all of the information it needs for configuration and code generation directly from the database, and it can generate the embedded code without manual interventions. This enables immediate transfer of changes made in modeling to the ECU code. This process prevents errors such as incorrect configuration of the code generating tool and guarantees error-free and complete code generation. This process also simplifies verification of the total system, since sections of the software have already been checked. It is possible to reuse the communication data for analysis tools like CANalyzer.J1939 or test tools like CANoe.J1939 from Vector, supporting the development of application layers.



[Figure 2: Code for the ECUs is generated based on the description of the electronic structure in eASEE's functional data management.]

The Volvo Truck Corporation chose a strategy for software development that has now become established in the automotive industry too: the use of AUTOSAR and its overlying tools (Figure 3). The benefits of this approach lie in the use of standardized and proven tools. They offer benefits in a development used intensively across brands that is distributed among many business sites. Common understanding of the underlying software structures and architecture is quickly achieved. It is easier to integrate suppliers, and it is not absolutely necessary to specify tools. This reduces dependencies on individual tool producers and suppliers.



[Figure 3: When standardized data formats are chosen, standard products can be used to describe and create the ECU-specific software.]

Problematic in this approach is the use of communication methods that are either incompatible with AUTOSAR properties or can only be used with it in a proprietary way. The use of J1939, in particular, should be mentioned in this context. While AUTOSAR essentially assumes a network of known nodes – and therefore a communication matrix that is known at the time of integration – this is decidedly not the case for J1939 with its plug-and-play concept. Volvo Trucks confronted this challenge with a two-prong approach. The first step was to identify which parts of J1939 were used in Volvo vehicles and integrate them in the existing Vector AUTOSAR tool chain. Secondly, Volvo – together with Vector and other European heavy-duty vehicle manufacturers – adopted portions of the J1939 protocol in AUTOSAR. This strategy lets Volvo exploit the advantages of AUTOSAR directly and universally. On the other hand, AUTOSAR integration of J1939 makes it possible to achieve fundamental independence in tool selection. Volvo chose Vector as its supplier of tools and embedded software components, since Vector already offers solutions in all areas, and it was possible to adapt them to Volvo-specific requirements very flexibly.

Revised: 11/2008
Word count: 1,086
Character count: 8,645

Figures:
Figure 1: The MAN Common Engineering Data Backbone
Figure 2: Code for the ECUs is generated based on the description of the electronic structure in eASEE's functional data management.
Figure 3: When standardized data formats are chosen, standard products can be used to describe and create the ECU-specific software.
All figures: Vector Informatik GmbH

Literature references:
[1] J. Svensson, "The Use of AUTOSAR in Volvo Group", presentation at Vector J1939 User Day; slides may be downloaded at: www.vector-informatik.de/j1939ud [most of them are German]

Author:

Peter Fellmeth is a team leader and product manager at Vector Informatik GmbH, where he is responsible for the development of products and customer-specific projects related to J1939, ISOBUS, Ethernet and DeviceNet.

Vector Informatik GmbH
Ingersheimer Str. 24
70499 Stuttgart
Germany
www.vector-worldwide.com

Editorial contact: Holger Heit
Tel. +49 711/80670-567, Fax +49 711/80670-58567,
E-mail: holger.heit@vector-informatik.de