Get a grip, get your system architecting right

In an ever more complex reality, you need advanced system architecting approaches to get a grip on that reality. Modern system engineering has too many dimensions for one brain to contend with, in an environment that is too complex to grasp without abstraction. Explicit concepts, supported by models are the in-between step, providing a clean dashboard, an understandable representation of reality as the starting point for the system architect's actions. It leads to the necessary overview and consistency to act and decide effectively.

Embedded Control Units running software have been integrated for about twenty years in trucks now and are becoming increasingly important. Raymond Tinsel is Supervisor and Lead Architect at DAF Embedded Engineering and gives a few examples: "About thirty Electronic Control Units control five hundred functions. Driver Assistance Systems, the DAF Connect information system, security, V2X connection and electrification, among other things, see to it that software becomes a larger chunk of the truck and our brand identity. Our latest model realizes a 7% fuel reduction, for a significant part thanks to software. Since 1993, exhaust emissions have been brought down by 95% - with the support of software."

As the 125 embedded developers have become involved in more and more domains, an organizational split between 'engine embedded' and 'vehicle embedded' has been made to keep better oversight. "Apart from that, a model based approach has been introduced to have only one single source of truth – including traceability and consistency functionality – during the entire process from functional design to implementation", Tinsel says. "We closely collaborate with TNO-ESI to apply as structured an approach as possible to further improve our methods. Reference architecting, reasoning methods and design patterns are deployed and made to measure to demands in our own domain."

Overview and consistency

After this introduction, Richard Doornbos from ESI reveals more about this Model Based System Architecting. "Continuous product evolution and series of one cause complexity to get out of hand. On top of this comes the introduction of autonomy and interconnection, making a many system part of a System of Systems. And then there are the 'x as a service' concept and the exploding amount of parameters to complicate things further. At the centre of all this development is the system architect, the spider in the web between business, engineers, sub system architects and lead designers. More and more concerns are loaded upon his shoulders, creating a quite disturbing picture of all his responsibilities: too much for one brain." More so, because there is uncertainty involved in many roles, caused by too many dependencies. "How to safeguard consistency while everything constantly changes?", Doornbos asks his audience.

In such a situation, Model Based System Architecting enables co-operation in teams and a systematic approach, regulates ownership and stakeholder dependencies, creates transparency and introduces abstractions to balance things and provide a grip on complexity. "One possible approach is to make a chart of all sorts of relations and dependencies with traffic lights in between. When disparate parameters such as 'societal acceptance', 'cost', 'noise' or 'throughput' make a connection between domains out of agreed 'contract' bounds, the traffic light turns red. Then, another trade-off, i.e. another balanced path through the chart has to be pursued. Manually, that would be a quite crude approach, but wherever possible, values can also be calculated. It is crucial and very valuable that every stakeholder involved uses the same definitions for parameters. The interdependence in the dashboard see to it that if one value changes, all other connected values change with it. In this way, overview and consistency go together. Consequences of decisions by the system architect become systematically apparent, enabling a balanced trade off over various domains."

Doornbos is enthusiastic, but has a warning: "The introduction of a model based approach requires investments and a culture change, which costs time. Decision on investments are sometimes made by senior managers who are not very deeply involved in system architecture. So when you plead for investment, make sure the benefits are made visible."

A need to adapt

System Architect Bruno van Wijngaarden from Vanderlande provides the perfect illustration from practice of how growing complexity demanded a totally different, more systematic approach. Vanderlande is a big name in warehouse automation and material handling systems and moved from project engineering to product platforms. Van Wijngaarden entitled the tale of this transformation 'Tales of a shepherd dog'.

"The project engineering approach, starting each project from scratch based on the customer's system requirements, became too expensive and didn't make efficient use of our available human capital. We needed to adapt", Van Wijngaarden explains. "This being said, you have to choose your new approach wisely and I think we did. Defining projects on the basis of product platforms not only results in more efficient and effective human capital deployment, but also in much faster order lead times."

In the project engineering approach, a system was defined in layers: material handling mechanics, low level controls and high level controls. Van Wijngaarden: "The product platforms approach focuses more on autonomy and System of Systems. The architect has in important role: he uses a solution tree to find the system solution the customer needs by scoring solution options against the customer's Key Performance Indicators: a customized system that meets the requirements built from selected standard solutions."

Domain driven

Design now becomes domain driven instead of the layered mechanics, low and high level controls approach. There is distinction between a business domain, a logistics solution domain and a materials handling domain - all independent from each other and all equally important to the success of the system. The domains provide an unambiguous, functional view of the system instead of a context-dependent view, with a constant awareness of the individual application. "All in all, it's a completely new paradigm", Van Wijngaarden concludes.

One of the first products developed with the new approach is the Adapto standardized/configurable system for storage and retrieval, a self contained system built by one team and with mechanic materials handling, low level controls as well as high level controls. It is flexible enough to serve as wide a range of customers as possible.

The project scope has shifted from system wide to the Self Contained System only. Van Wijngaarden: "The system architect has the task of a border collie to safeguard the new boundaries that have become important. He has to walk the perimeter to see to it that the herd goes in the right direction. Also, as a System Architect he has to balance between Sales that tends to regard the new approach with platform development as too rigid and Product Development that regards it as too flexible."