SIEMENS

Ingenuity for life

Siemens Digital Industries Software

# V-cycle strains as cars morph into consumer gadgets


## Executive summary

The consumerization of the car is pressuring carmakers to have shorter, more flexible design cycles. This is problematic for design teams using the traditional approach to the V-cycle, usually composed of disconnected processes that make it especially difficult to respond to the gadget, quick-turn trend. What's needed is a controlled and seamless flow of data across departments – also automation and abstraction of requirements such as auditing changes and modeling new functionality, both in terms of correctness and cost. And ideally, design tools should link all the way to the creation of service documentation, a major challenge given the exploding number of vehicle configurations. The main task, then, in this new connected-car era is building more supple, flexible design flows better able to adapt to changes in customer preferences more apt to be shaped by smartphones than engine specs.



Sjon Moore
Technical Marketing Engineer
Siemens Digital Industries Software

# Ever shrinking cycle times

The average consumer is surrounded by gadgets. It's rare to find someone who doesn't have a smart phone in their pocket. Notebooks, tablets and smart watches are always nearby for ever increasing numbers of people. Though just how healthy it is to be constantly connected is an open question, we undeniably do live in a 'gadget culture' where people want the latest and greatest technology – in some cases, because it legitimately improves their lives, and in others just because they want it.

The cycle times for consumers to upgrade or refresh their gadgets is generally shrinking. The average consumer upgrades their mobile phone at around 29 months. This has increased slightly over the last year, but it's still a very short amount of time if one considers the development cycles of an automobile.

If the mobile phone (or any other device for that matter) is updating on a short cycle time, but the automobile is not, consumers may find themselves in situations where their new phone won't work with their 'old' car. Another developing technology that's likely to drive shorter cycle times is connectivity to 'things' outside the vehicle such as infrastructure or even other vehicles. As these technologies deploy, they'll be seen as 'must haves,' especially for first-time car purchasers who have always lived in a world with the internet, and who have rarely used a land-line phone, and thus has very different expectations of what an automobile should be. Their expectations of a thing's usefulness is defined by what they can do with the internet-connected gadgets they've grown up with.

# The main task for OEMs

There is already a pipeline full of new technologies and advanced features that are being readied for deployment into the automotive market. The market pressures described above will drive further pressure on this pipeline to get it done faster. But, as always, cars must be competitively priced so costs must be kept in check, and no one can afford costly warranty or recalls. How does an OEM ensure that their engineers can safely and cheaply deploy the features in this pipeline as quickly as possible? As I see it, a fundamentally different approach to engineering is required.

Consider the traditional 'Systems V' design cycle where requirements flow from the voice of the customer at the top left, cascading down through levels of systems, sub-systems and ultimately components, all to be validated as they're built up on the right side of the 'V'. If time forms the x-axis, then the pressure from the market is exerted from the right towards the left (see figure 1).

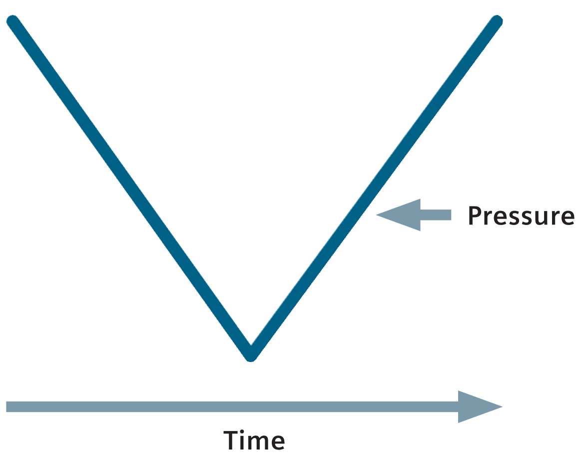


Figure 1: Market pressure, particularly the expectations for shorter design cycles, squeeze the ideal System V flow.

In traditional engineering flows, this 'V' is rarely as connected, continuous and coherent as shown above. It's normally held together by disconnected processes spanning multiple departments using their preferred point tools. Data that needs to cascade from one group to the next is either manually re-entered, or an ad-hoc export/import through an intermediate file which might be transferred via e-mail, stored on a network drive, or simply passed via a USB drive. In reality, our 'V' ends up looking more like islands of activity (see figure 2).

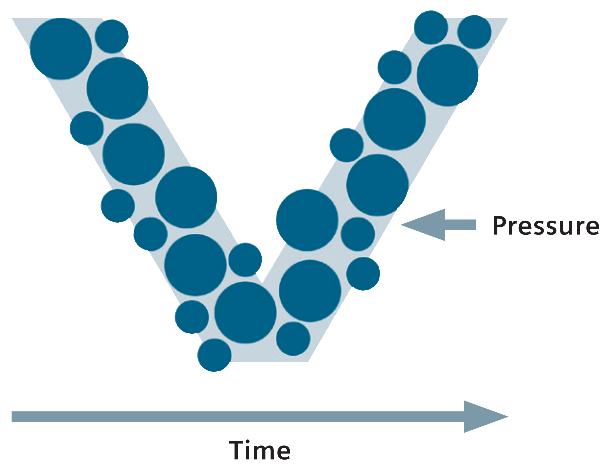


Figure 2: In reality, most V-cycles are composed of disconnected processes and tool, making the overall cycle brittle to change.

An engineering process and data flow such as this will not remain intact as the pressure mounts. The obvious failure points will be the hand-over points between these islands of engineering activity. What is needed instead is a coherent process where data flows seamlessly between organizations (see figure 3).

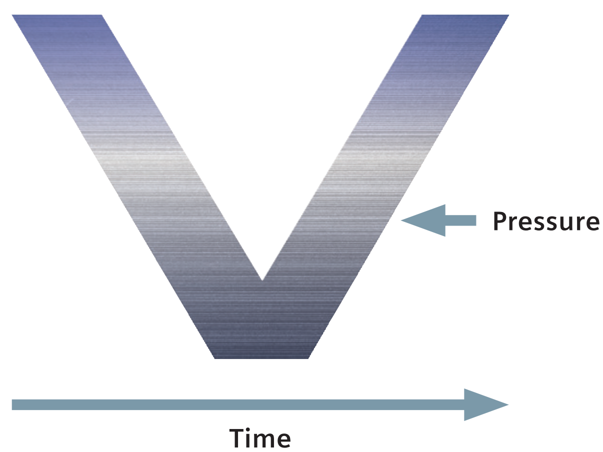


Figure 3: Seamless flows of data down and up the V-cycle strengthen the overall design flow, and make it more resilient to change.

What's required is a tool suite that manages user data without any manual transfers which compromise data integrity, expose the data to errors, and enable a critical failure point in the process. This doesn't mean that all engineers must have the same user interface and do things the same way. In fact, each engineering organization has very different use-cases, business drivers, and goals. As such, the tools they use should reflect that. When the System V is strengthened by such a coherent tool and method, it will more easily shrink under the market pressure while maintaining solid engineering integrity (see figure 4).

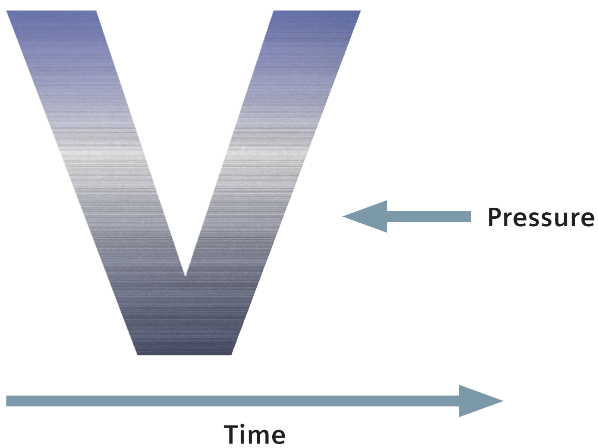


Figure 4: Coherent design methodology allows the V-cycle to flex as the overall design process speeds up.

Any engineering process will be bombarded with a steady stream of changes. Managing these changes has always been important, but will become increasingly so as cycle times shorten. Having a coherent process where data flows seamlessly is critical, but it must be done in a controlled manner. If the data were always left wide open for changes to occur at any time, chaos would ensue. Simple things like designs having release statuses, controlling user permissions, maintaining an audit trail of who did what when, and tracking changes through change documentation are all basic and mandatory requirements.

But, to truly enable rapid deployment of new functionality, it is necessary to inject this functionality at the right point in the System V, then synchronize this change to the rest of the design organizations in a controlled fashion.

For example, assume that the top left of the process starts with functional modeling, which is then applied to a platform architecture, which results in logical system designs, which are applied to a platform physical topology, which results in physical wiring, which is synchronized to harness designs, which are broken down into harness assembly processes, which ultimately results in an actual harness product being built. Let's also assume that all of this data is re-used by a service organization to creates service documentation and troubleshooting procedures for their technicians in the field (see figure 5).

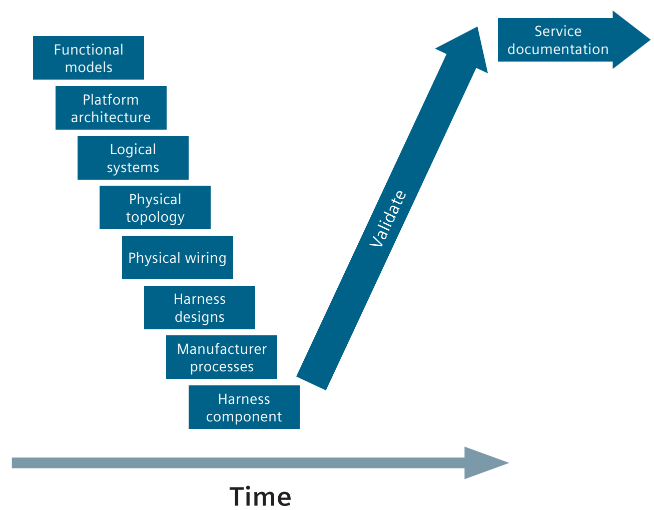


Figure 5: New functionality should cascade through the V-cycle, from initial modeling all the way to service documentation.

What if the technology the market is pushing for is an off-the-shelf design from an established supplier and is to be deployed into an existing platform? Is a complete breakdown of functional models required to deploy this correctly? Is a complete re-assessment of the platform architecture required? Perhaps the supplier has already provided logical system designs (remember, this is off-the-shelf). The quickest way to deploy such a system would be to inject the new system designs into the process, then cascade it down.

But, this would mean the functional models and platform architecture would be out of date, and would no longer match the actual released design. This can be an issue if the next market-driven functionality requires a full assessment of the functional design. How is this possible if the functional and architecture designs don't match the production design?

What's needed is an ability to synchronize data from one abstraction to another in a controlled way. As the downstream tasks of developing the wiring and harnesses for this new system are underway, the organization responsible for functional modeling and architecture design can begin synchronizing their designs to the logical system designs that were injected. By doing a cross-abstraction comparison, they can see what would need to change in their architecture and functional models to be consistent with the new changes. They can either make the modifications manually, or ideally use tools that do this automatically. Once this is complete, all data within the process is coherent and consistent but didn't require the extra time required to start all the way at step 1.

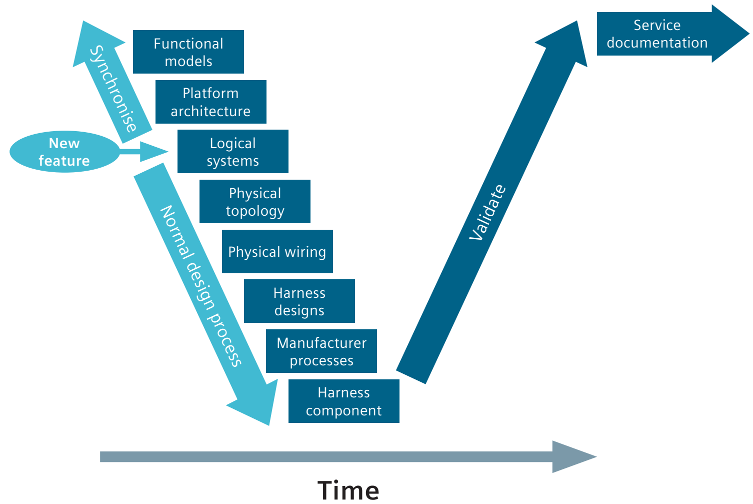


Figure 6: V-cycle should accommodate new functionality being inserted at various points along the flow, and then synchronize appropriate changes up/downsteam.

# Enabling technologies

Manual design processes are slow and error-prone which are contradictory to two aspirational attributes we're trying to achieve, speed and correctness. As much as possible, design processes must be automated to remove the engineer from the critical path of inconsequential decision making. Instead, the engineer should be working at a higher level to populate the rules that automation will use to make the thousands of tiny decisions. These rules can then be re-used throughout an organization to ensure consistency in design quality.

**Siemens Digital Industries Software**

**Headquarters**
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

**Americas**
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

**Europe**
Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

**Asia-Pacific**
Unit 901-902, 9/F
Tower B, Manulife Financial Centre
223-231 Wai Yip Street, Kwun Tong
Kowloon, Hong Kong
+852 2230 3333

## About Siemens Digital Industries Software

Siemens Digital Industries Software is driving transformation to enable a digital enterprise where engineering, manufacturing and electronics design meet tomorrow. Our solutions help companies of all sizes create and leverage digital twins that provide organizations with new insights, opportunities and levels of automation to drive innovation. For more information on Siemens Digital Industries Software products and services, visit siemens.com/software or follow us on LinkedIn, Twitter, Facebook and Instagram. Siemens Digital Industries Software – Where today meets tomorrow.

**siemens.com/software**



77778-C7 4/19 H