ERTS² 2012 – Paper Submission

Title: Systems Engineering for Cyber-Physical Products

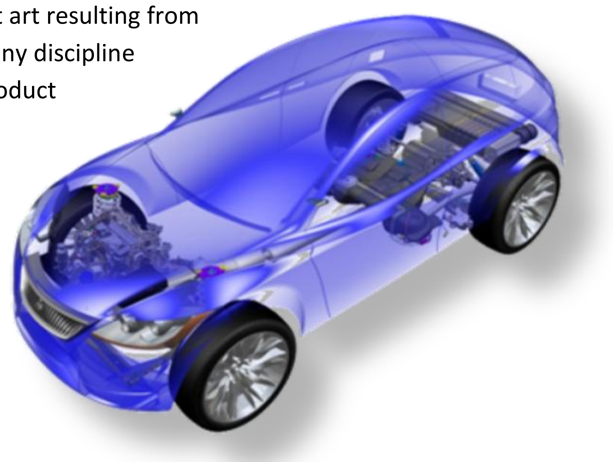
Abstract: This paper will present how the Dassault Systèmes PLM solution introduces a new paradigm to address the systems engineering challenges of developing cyber-physical systems. V6 unified modeling architecture has extensive support for cross discipline systems engineering based tools, enabling a collaborative Platform and Model Based Engineering environment.

Author: Bernard Clark (Dassault Systèmes) [Bernard.Clark@3ds.com]

Keywords: MBSE, cyber-physical, architecture, modeling, simulation, embedded systems, MODELISAR, MODELICA, FMI, FMU, co-simulation

Table of Contents

Introduction and Scope of the Problem



Systems engineering of complex cyber-physical products is a difficult art resulting from multiple collaborative multi-disciplinary and iterative processes. Many discipline owners across multiple organizations, participate throughout the product lifecycle to understand the needs, develop requirements, evolve the product's functional and logical architectures while optimizing constraints, develop the individual solution components, and then integrate and validate them through simulation of virtual assemblies and system behavior. This creates an increasingly difficult layer of complexity as more features are added to the product and need to become tightly integrated with other systems within the product. The result is a complex web of interrelated systems where a problem in one strand can devastate the functionality of the others.

In general, industry is convinced of the benefits of 'Systems Engineering' as a means of developing optimal solutions (design to performance). However, the benefits derived fall short of general expectations due to the inability to really create collaboration and orchestration of the different technologies and disciplines involved in the business processes. In conjunction, the lack of control in managing data and model integration consistency across a large number of engineering tools – particularly in the context of highly configured products developed across an extended enterprise, continues to represent a significant challenge for most organizations.

The problem stems from the classical systems engineering approach, where it is difficult to integrate results of poorly connected tools in the systems engineering development process. It is not uncommon for organizations to literally use hundreds of different tools at various stages of the systems development process.

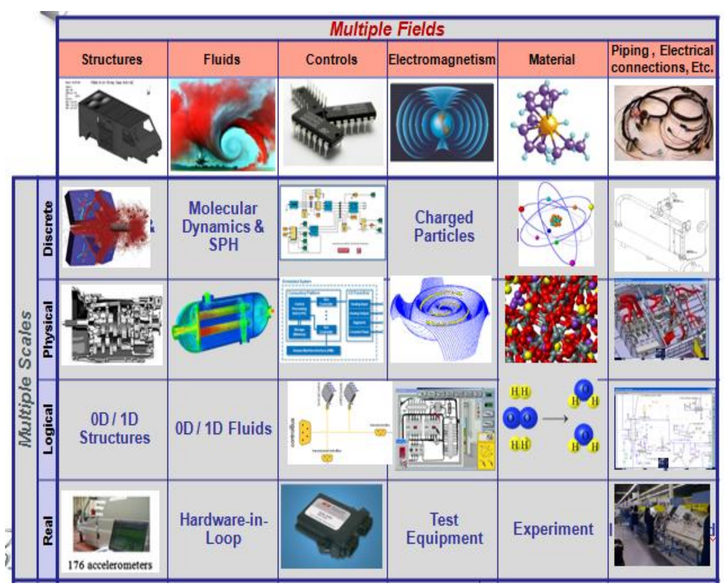
The crux of this problem stems from the typical approach to defining systems architectures based on descriptive and not simulation capable tools. Today, examples include using a combination of UML & SysML (or even Visio) based modeling tools.

In the context of cyber-physical products, while these higher abstractions are all excellent for the high level systems architecture definition and detailed design distribution of a given software module, they are deficient in their inability to be the center for design integration enabling engineers to spot faulty interactions in the overall systems and product development process.

The Cyber-Physical Challenge

To deliver a complex cyber-physical product involves leveraging different modelers at different stages of the overall development and production support process:

- The system composition models used to create the accurate ***physical, manufacturable product*** – usually called Digital Mockup (DMU) – are created with 3D geometry modelers such as mechanical Computer Aided Design (CAD) systems. DMU is a complete 3D geometrical composition that enables digital component assembly and management.
- The system composition models are also used to create the accurate ***dynamic and performance representation*** of the product. These models are typically defined by mathematical equations that can be composed to perform simulations of the virtual behavior of the product. The models can be tightly coupled between engineering domains with the ability of being further refined to give real-time results. They are often used as the basis for driving / flight simulators, or for hardware / software validation platforms like hardware-in-the-loop (HIL) systems.

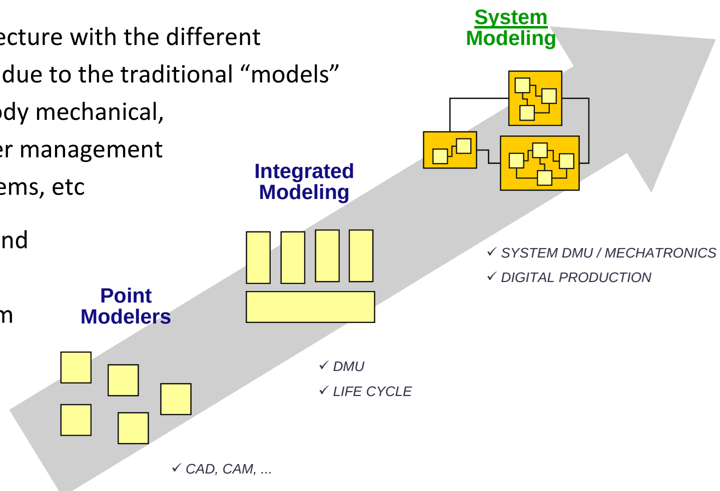


- Finite elements analysis models are used for the computation of ***the accurate limits of each part or sub-assembly composing the product.*** They are done through solving multi-physics based detailed models – usually called computer aided engineering (CAE) models – that enables the computation of stress, thermal limits, fluid flow, part interactions, etc., from different levels of accuracy of the finite element definition and composition.
- The control models are used to create the control algorithm for the control systems that will ultimately interact and operate the mechatronic or cyber-physical systems.

- The composition models that are used to create the accurate model of the ***digital embedded computing platform*** – usually called the model of computation and communication (MOCC) – models the embedded and power distribution systems. These models have the ability to integrate networks, computing nodes, sensors and actuators on real time execution and mode management platforms, and enable sub systems suppliers to create components that can be progressively and accurately integrated together.

The challenge today, with using these different modelers, is that there is:

1. No ability to provide model integration into comprehensive – functionally accurate and simulation capable – but functionally abstract solutions.
2. No ability to have configuration management of the systems architecture at the granular level of an 'entity', making it almost impossible to apply product line engineering principles.
3. No ability to share the systems architecture with the different engineering domains in a unified way due to the traditional "models" leveraging schematics of the multi body mechanical, hydraulic, pneumatic, electrical, power management systems, control systems, sensor systems, etc
4. No ability to quickly and easily map (and maintain linkages) from the entities between systems architecture diagram through to the instantiation of the entity in the global & accurate virtual product.
5. No integration capabilities exist between the high level product requirements definition through to the decomposed functional, logical and discipline specific architecture models, and then through to the instantiation and simulation of these models in the global & accurate virtual product definition.
6. No integration between the embedded controlling development process and the global & accurate product modeling environments.

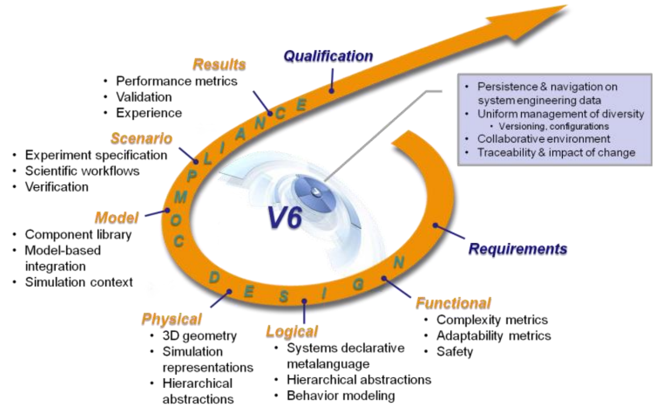


Engineering teams developing complex cyber-physical products are demanding a more unified and integrated approach to Systems Engineering[1]. They need an open platform that is capable of support libraries of components that can be composed into systems models. Ultimately these components need to be configurable so that different behaviors of proposed systems can be readily simulated and analyzed. Systems engineers also need tools that enable them to quickly and easily define and navigate the complex relationships that exist between the many different entities that make up the complete product with all of its embedded systems.

The Dassault Systèmes V6 Systems Engineering Platform

The Dassault Systèmes V6 platform unifies the design (Requirements-Functional-Logical-Physical) and compliance (Model-Scenario-Results-Qualification) processes, in an environment that inherits the core values of the V6 collaborative integration architecture, such as persistence and navigation on system engineering data, uniform management of versioning and configurations, traceability, and impact analysis of change. Furthermore, the integration architecture combines best-in-class tools in a managed innovation environment that provides a next-generation approach to systems engineering.

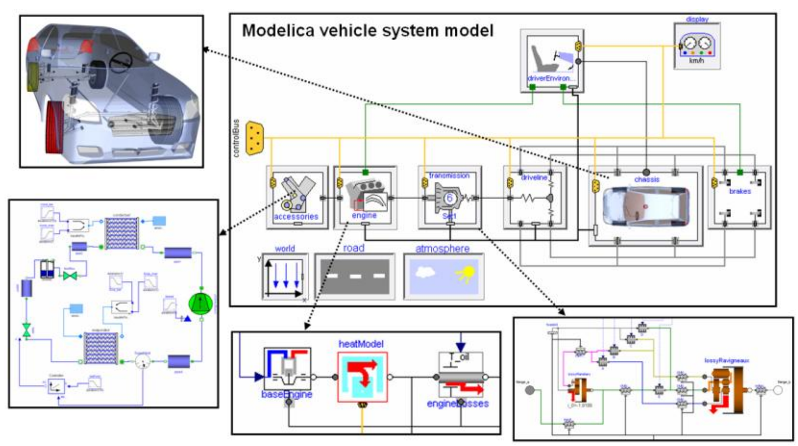
The solution outlined is rapidly gaining acceptance as it enables a spiral of innovation, based on a continuous digital chain of elements that have the proper semantic to be traced and leveraged for impact analysis, design change and product line flexibility.



Dassault Systèmes – Innovation Spiral

The Role of Standards

To address the cyber-physical systems design, modeling and simulation challenges outlined earlier, Dassault Systèmes initiated two successful European standards initiatives (EuroSyslib and MODELISAR), and invested heavily in developing an integrated tool set to support this systems engineering needs. These tools offer an open and extensible system engineering development platform and fully integrated cross-discipline modeling, simulation, verification, and collaboration environment. Both of these projects leverage the Modelica language.



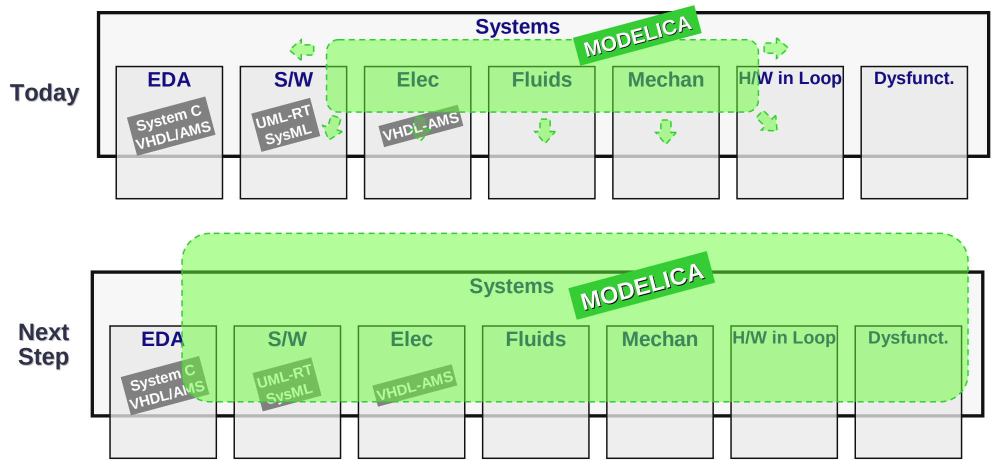
Modelica is a relatively new language that offers a robust solution to address the needs of industry brought about by the increasing complexity of products and systems, and the need to improve quality and reducing overall time to market of these complex products.

Modelica is defined and managed openly, with the objective of delivering a scalable, equation based, dynamic modeling environment that unifies multiple engineering and physics domains. By leveraging investments in component libraries created using the Modelica language, it provides the ability to design, optimize, and check, as early as possible in the design process, the behavior of a planned future product in a virtual environment.

Modelica is designed to solve difficult system problems, for dynamic interaction giving performance estimates and measurements in particular:

- Multi-discipline problems involving simultaneously technologies from ***multiple domains*** such as: mechanical, hydraulics, pneumatics, thermodynamics, flow dynamics, electrical, software, real-time, etc.
- Problems where the components are ***highly coupled*** together, where traditional hierarchical design does not work, or does not readily provide the ability to reach optimal designs
- Problems involving hybrid mathematic solving such as continuous-discrete modeling and simulation
- Discontinuous and variable structure systems

Modelica has the potential to become ‘the’ standard for dynamic system modeling, in all disciplines.



EuroSyslib, was a project initiated by Dassault Systèmes whose experience showed that it is possible to work with an open language that properly integrates all disciplines, and that that this language provides an acausal, very powerful, mathematical based solver to support all systems / physical domains.

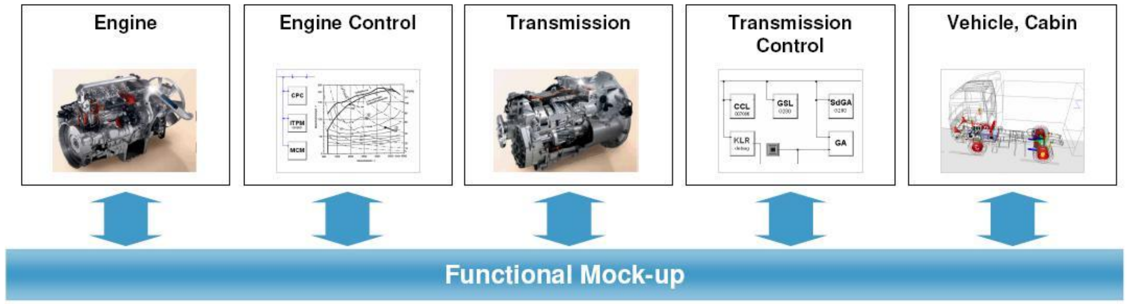
MODELISAR was a European ITEA2 research project, initiated by Dassault Systèmes and Daimler, with the main objective being to boost collaboration and innovation across system and software disciplines through the integration of system & software simulation at the complex vehicle level. The MODELISAR objective of enabling early vehicle performance and behavior tests in the virtual world, and ensuring seamless and traceable product development. To make it practical, the MODELISAR project was focused on connecting the Modelica and AUTOSAR standards.

The MODELISAR project started in July 2008 and was completed in December 2011 with a total funding of €27M.

MODELISAR leveraged the ability to provide open model integration and co-simulation between virtual product models, as well as Modelica capabilities. It also focused on supporting widely used models in proprietary formats (e.g. Simulink, etc.) and the ease of integration of these models for other levels of virtual execution of embedded software, under various configurations.

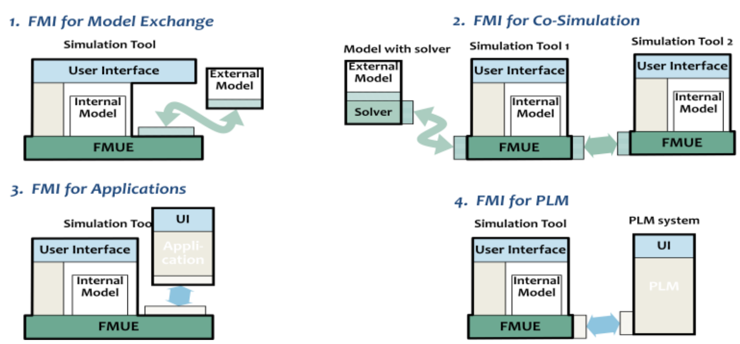
The outcome of the MODELISAR project is a new open standard model exchange and co-simulation framework, targeted at the class of problems expressed above, called “Functional Mock-up Interface (FMI)”.   This standardized interface supports exchange of models that are described by differential, algebraic and discrete equations with time-, state- and step events.

FMI provides advanced runtime interoperability interfaces that enable accurate model compositions to be created by allowing several pre-compiled simulation units to be combined into one simulation framework.



FMI is designed to be an open, general, vendor independent tool interface standard for enabling systems simulation. The FMI specifications are published under a copyright free license. It includes the definition of four key capabilities for model composition including model interface, co-simulation interface, lifecycle management interface and application interface (including HIL). It also included 15 proof-of-concepts on different disciplines, five on code, HIL, calibration and test, and five on lifecycle management and the ability to compose heterogeneous models. The participants included large conglomerates such as Daimler and Volvo, large research facilities (e.g. Fraunhofer Institute and Wittenberg University), as well as software tool vendors and SME's.

The value of FMI is that the export formats generated can be "composed" – manually with very lightweight tools – with non models coming from other (non-Modelica) environments.



These composition capabilities are expressed in the drawing below in an automotive use case where often specialized or legacy tools are used to create subsystems.

In practice the FMI standard has four layers and is implemented through a standardized XML description that acts as meta-data to enable the digital composition. The specification can be downloaded from www.MODELISAR.com/fmi.html .

The standard for Model Exchange

The intention is that a modeling environment can generate C-code of a dynamic system model that can be utilized by other modeling and simulation environments. Models are described by differential, algebraic and discrete equations with time-, state- and step-events. The models to be treated by this interface can be large for usage in offline or online simulation, or can be used in embedded control systems on micro-processors. It is possible to utilize several instances of a model and to connect models hierarchically together. A model is independent of the target simulator because it does not use a simulator specific header file as in other approaches. A model is distributed in one zip-file called FMU (Functional Mockup Unit).

The standard for Co-Simulation

The FMI definition provides an interface standard for coupling two or more simulation tools in a co-simulation environment. The data exchange between subsystems is restricted to discrete communication points. In the time between two communication points, the subsystems are solved independently from each other by their individual solver. Master algorithms control the data exchange between subsystems and the synchronization of all slave simulation solvers (slaves). All information about the slaves, which is relevant for the communication in the co-simulation environment is provided in a slave specific XML-file. In particular, this includes a set of capability flags to characterize the ability of the slave to support advanced master algorithms, e.g. the usage of variable communication step sizes, higher order signal extrapolation, or others.

The standard for component management

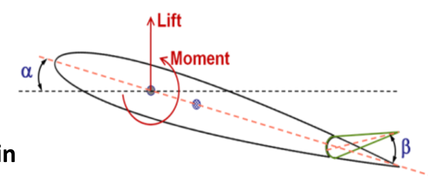
The intention is to provide a generic way to handle all FMI related data needed in a simulation of systems within a "Product Lifecycle Management" system. This includes:

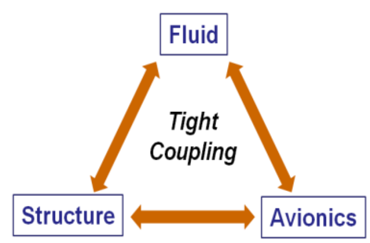
- **Functional Mock-up Unit data**, needed for: editing, documentation, simulation, validation;
- **Co-simulation data**, needed for: editing, simulation, and results management;
- **Result data,** needed for: post-processing, analysis, report.

Generic processes are defined here, as well as a format description to communicate between the PLM system and the authoring tools.

In order to illustrate the application of FMI in the area of co-simulation, let us consider the design of an airfoil, where we have to optimize the design in the context of number of different solution areas by showing the Functional Mockups to perform co-simulation of multiple systems models coming from a number of different tools.

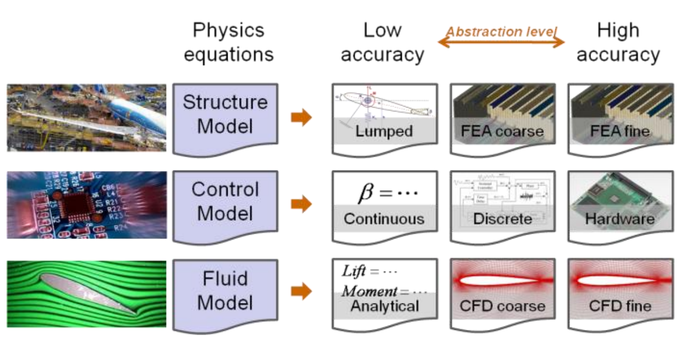
An airfoil with flap control illustrates the complex behavior of a product (the physical or 3D product is the airfoil, the cyber product is the flap actuation control) in context (fluid flow).





The need for Multi-disciplinary Design Optimization

An ever-increasing drive to improve performance, reduce costs, and increase efficiencies associated with complex system development has led to the need to explore computational methodologies that enable the development of better systems in less time with higher quality and reliability. This impetus has been particularly visible in industries where the complexity and multidisciplinary aspect of systems can lead the design team to challenging problems involving conflicting requirements that do not appear to have an optimum solution space. Two of the most important computational methodologies required are multi-physics simulation and multidisciplinary design optimization (MDO).

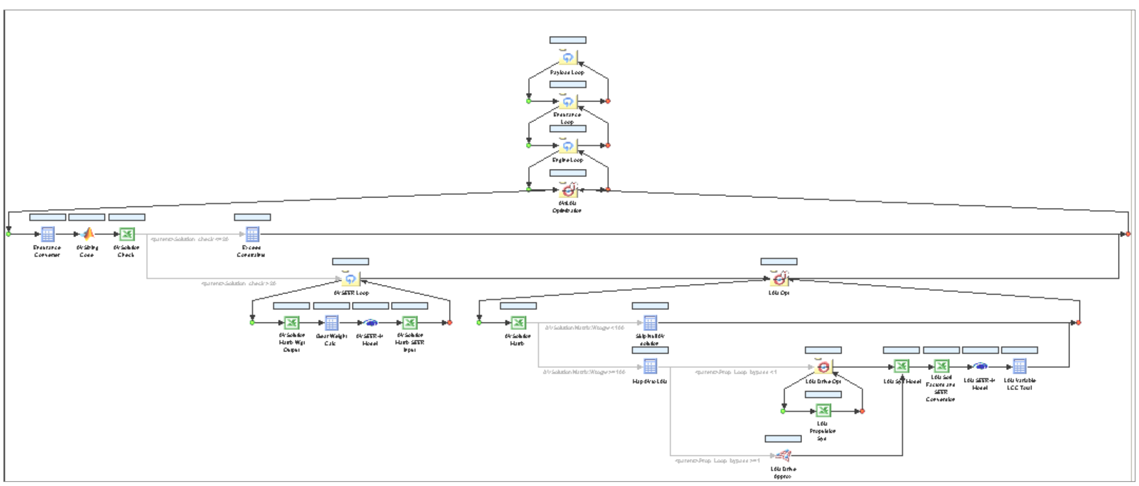


Multiple hierarchical abstractions in the RFLP-based design product structure are available for use in simulation models.

Design interferences introduce additional challenges to modeling complex cyber-physical products, as changes to assumptions in one model can have cascading effects on others. If we consider an automotive example, where the performance of antilock braking systems provides a good case in point, braking distance can be shortened by increasing the size of the tires; however, bigger tires may in turn penalize fuel economy, increase vehicle weight, and decrease vehicle aerodynamics. In turn, such adjustments may also dictate changes in embedded software logic.

Multidisciplinary design optimization is a field of engineering that uses optimization methods to solve design problems incorporating a number of engineering disciplines simultaneously. Although including all disciplines simultaneously significantly increases the complexity of the engineering design problem, the optimum of the simultaneous problem is far superior to the design found by optimizing each discipline sequentially, since it can account for interactions between the disciplines. Dassault Systèmes's MDO products are built on Isight technology.

Isight is a software framework that replaces the manual trial and error portion of the traditional design process with an automated, iterative procedure. Isight loosely couples all of the relevant modeling codes then automatically runs these codes, evaluates the output, adjusts the input based on defined objectives, and reruns the codes, continuing with this process until the objectives are satisfied. Isight is able to integrate all relevant design requirements and meet all design constraints. Isight combines the power of process integration and automation with design exploration tools including multi-objective optimization, design of experiments, reliability and robustness, and Monte Carlo simulation.



Typical "System of Systems" Optimization Flow Represented in Isight

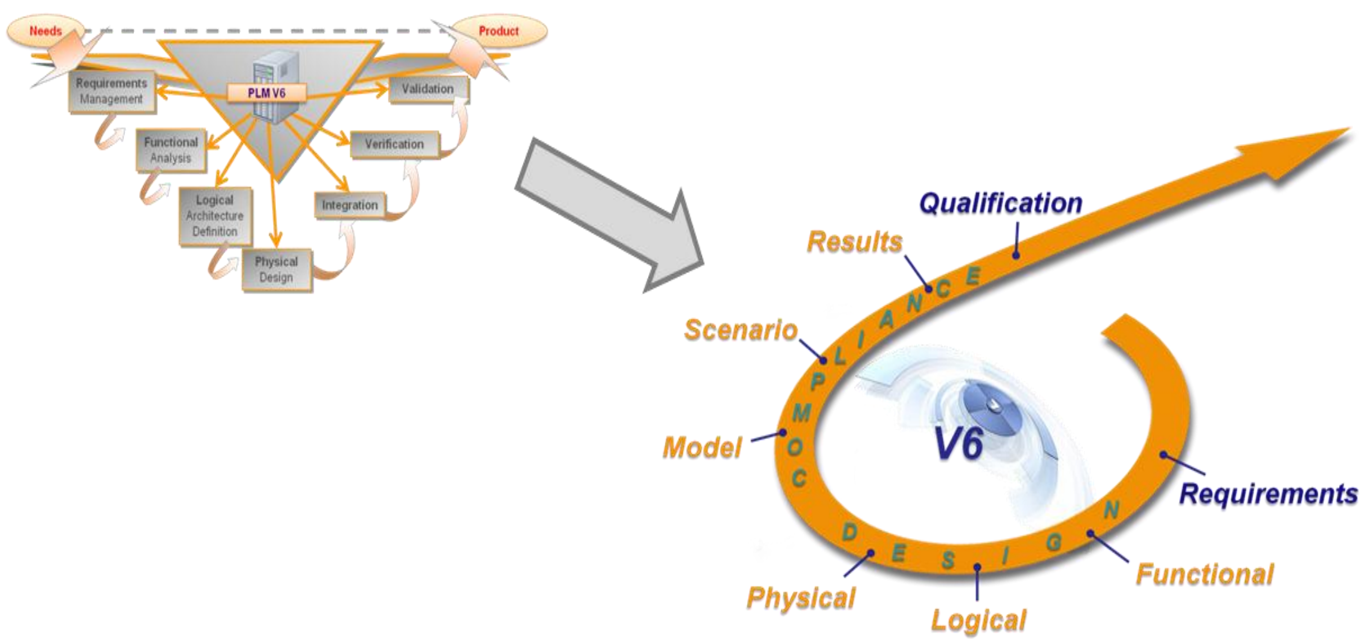
The widespread application of MDO is a testament to its power for close-coupling complex and often conflicting driving requirements in a modular design approach that enables development cycles to be compressed, ensure clear traceability of the design optimization path, and give the design team the ability to obtain improved optimum solutions.

The deployment of MDO can be very effective in systems-level design as a bridge between disciplines and subsystems. It can also be used as an optimization tool for exploration of design solutions when coupled with higher fidelity computer-aided engineering tools (i.e., finite element analysis in structural design or computational fluid dynamics in aerodynamics) and multi-physics tools involving coupling of multiple, high-level design disciplines (i.e., fluid-structure interaction problems or software coupled with electromechanical components). Examples of applications where coupling of MDO with high-fidelity computer-aided engineering design tools has been successful include manned aircraft structural optimization, race car design, and yacht design. For all of these applications, product performance is a main driver.

Managing the Systems Engineering Processes and Artifacts

The Dassault Systèmes PLM based V6 unified modeling architecture has extensive support for cross discipline systems engineering based tools, enabling a collaborative Platform and Model Based Engineering environment. This architecture provides:

- A rich and extensible data model and collaborative business process support environment. It provides a comprehensive engineering data management and collaboration environment, with requirements, platform, program, project, product, system definition and configuration management capabilities all derived from the same data model.
- An open Simulation Interface between tools for creating Functional Mock ups with global & accurate virtual product complexity in range with cyber-physical.
- An open modeling language already supported by two DS tools and five third party tools to enable modeling investment on parametric and complex systems to become possible inside and in collaboration with partners and Research communities.
- Together the support for definition and supports for the Model, Scenario, Result, and Quality modeling as the base methodology for systems validation, verification and qualification.
- Integration on PLM to defines the Process, Planning and Resource model for defining and validating product manufacturing, delivery, operation, maintenance and de-commissioning support.
- Provides full configuration management and lifecycle support for all artifacts produced throughout the lifecycle, while maintaining the integration capacity with traditional Embedded Software, MCAD, ECAD, CAE, physical modeling, simulation and control systems modeling tools.



From the 'V' model to a continuous spiral of innovation

Summary

The solution outlined provides a next-generation approach to systems engineering of cyber-physical products. It provides:

- Collaborative systems engineering development environment
- Persistence & navigation on systems engineering data, models, simulations and virtual experiences
- Uniform management of diversity with full versioning and configuration management of systems artifacts
- Traceability and impact analysis of all proposed and implemented changes
- Integration of legacy models & tools

The solution presented, with its rich and open data structure, the inbuilt collaborative business process support, and the fully integrated domain specific modeling and simulation environments, is unique in industry today. It enables the ability to quickly and easily evaluate requests for changes or new cyber-physical product or system variants, and offers better flexibility both in business terms and expected performance terms, leading to a unified performance based systems engineering approach and optimization of the cost of ownership.

References

1. Systems 2020: New Engineering Approach to Defense Systems, Final Report, Booze Allen Hamilton, 16 August, 2010
2. Using the Functional Mockup Interface as an Intermediate Format in AUTOSAR Software Component Development, Bernhard Thiele & Dan Henriksson, Modelica Conference, 2011