

## Highlights

- Separate design issues from those related to programming.
- Increase the productivity of engineers.
- Improve the capacity to scale and reuse components (models and modules developed).
- Incorporate the verification and validation processes in the early stages of development.
- Minimize the need for programming software code.
- Reduce integration time.

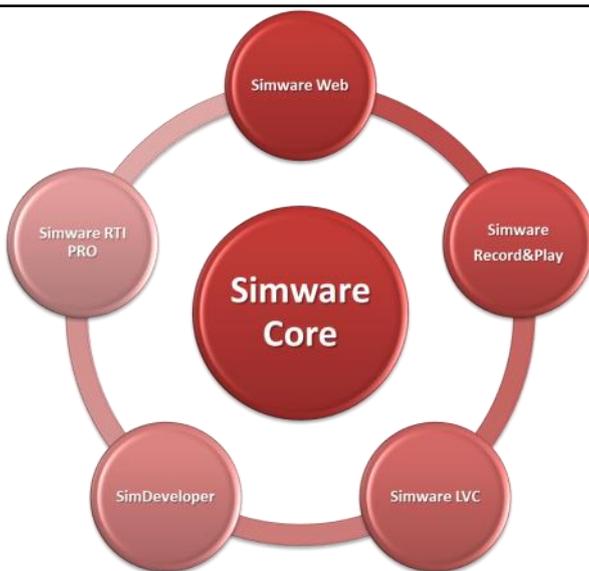
## The Engineers' choice

**SimDeveloper provides a Visual Modeling Environment based on Simulink to focus your engineering resources in the most important thing: modeling. Not coding. Say goodbye to unfinishable debugging and complex C++ code. This is an easy-to-use extension to SIMWARE CORE and enable the use of the Model Driven Architecture (MDA) paradigm with SIMWARE PLATFORM. Just what your Program Manager will like.**

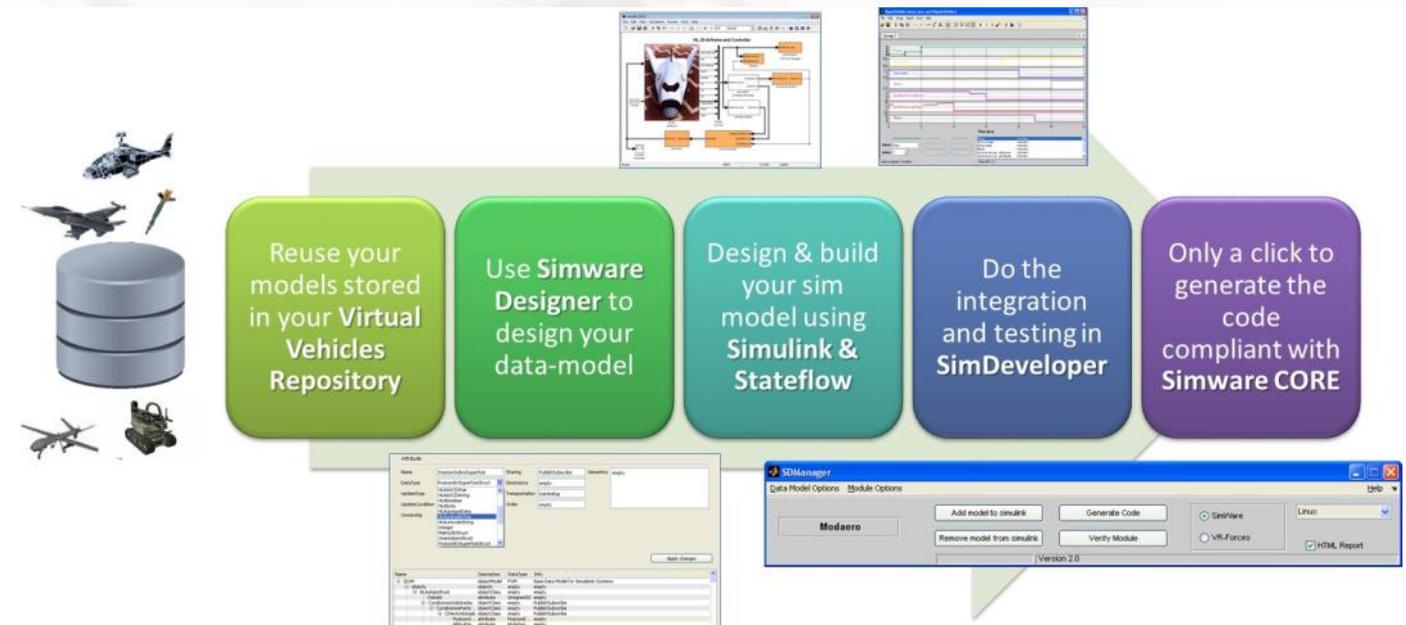
SimDeveloper is an environment or framework consisting of a set of commercial tools whose central axis is the whole development of the simulation modules. As such, it covers all areas related to the development cycle of software components (analysis, design, development and unit tests) to its integration with the rest of the systems.

SimWare SimDeveloper is based on the Model Driven Development (MDD) paradigm, providing a Simulink based GUI for the development of Simulation Assets. Because no computing skills are necessary, SimDeveloper provides a very high productivity development environment for M&S, systems and Subject Matter Experts (SME) engineers, minimizing time to market, reducing both cost and risk for the development of high-fidelity simulation models.

The use of SimDeveloper enables the **concept of M&S repositories**. The configuration element will be the engineering model, not the source code. Generated code is in **full compliance with SimWare Data-Centric and Open architecture**.



**SimDeveloper is the Model Driven M&S Development tool included in Simware Platform as an extension to Simware Core.**



SimDeveloper supports the whole life cycle of the simulation models for any kind of simulation solution, from a High-Fidelity Full Flight Simulator to the most complex constructive simulation.

## Engineers take back the control

There is a comment that states that the ability of engineers to define, analyze and develop physical problems, often contrast with their ability to implement effective software codes and vice versa, good programmers may in many cases lack the physics skills necessary to develop efficient engineering software codes. Because of that, the traditional development of simulation software models is not affordable, consuming a lot of manpower, using slow and prone to errors procedures.

In order to solve this challenge, NADS has developed a new way of creating simulation modules, we named it **SimDeveloper**.

## Leveraging Simulink to increase productivity

SimDeveloper is based on Mathworks® products: Matlab and Simulink and their associated code generators. It generates ANSI C++ code that has the clarity and efficiency of professional handwritten code. The generated code is exceptionally compact and fast, essential requirement for real time performance.

## Conceptual Shift

The SimDeveloper Methodology is oriented to connect the system engineer with the logical and mathematical modeling of simulation modules, hence placing it within an engineering environment isolated from the software code.

This conceptual shift allows to obtain the next advantages:

- Separate design issues from those related to programming.
- Increase the productivity of engineers.
- Improve the capacity to scale and reuse components (models and modules developed).
- Incorporate the verification and validation processes in the early stages of development.
- Minimize the need for programming software code.
- Reduce the Integration times.

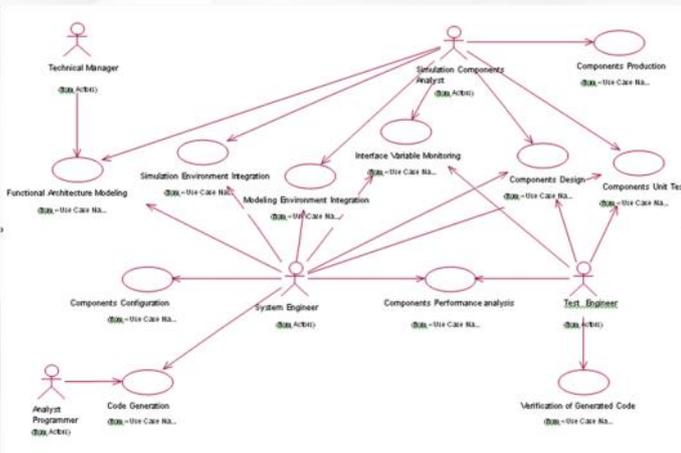
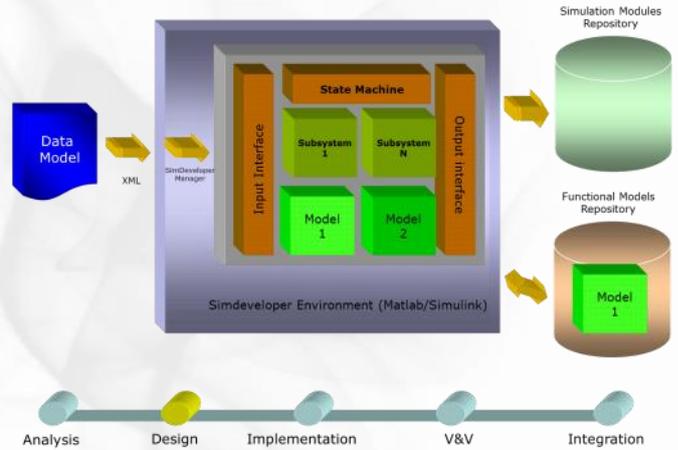
# The Code is Over



## A lean process

The process of development considered in SimDeveloper is composed by a set of tools that cover:

- Data-Centric design of the simulation models.
- Visual development of the models
- Visual Verification and Validation of the models with Unit and integration testing
- Automatic generation of C++ code compliant with Simware Platform
- Configuration management of the models.



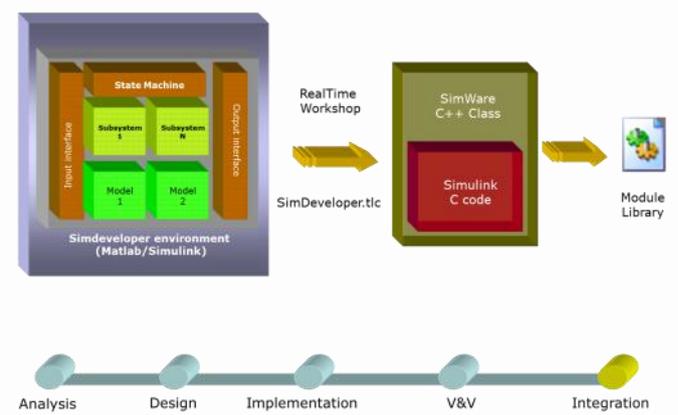
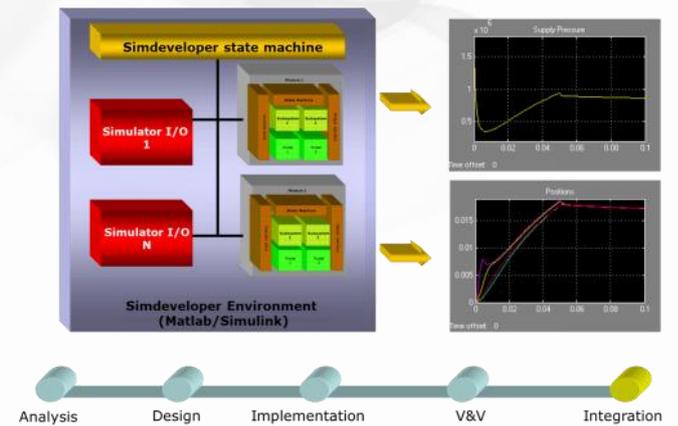
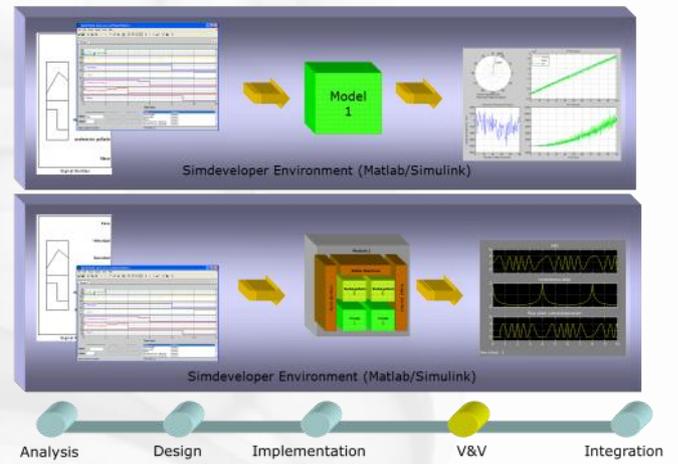
View of the subprocesses corresponding to the SimDeveloper

## Scope

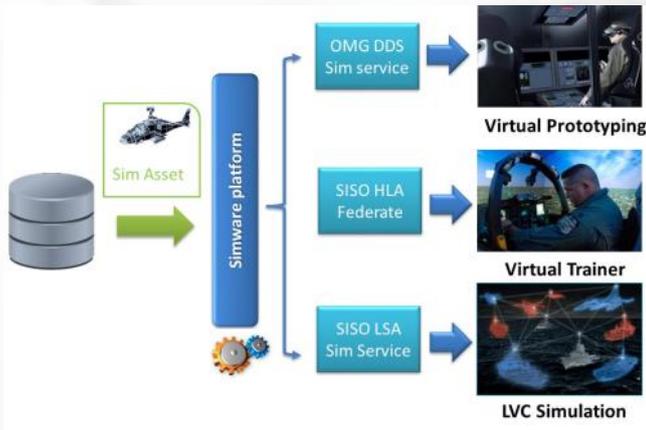
SimDeveloper covers the full design stage of the module in which, from an initial documentation that includes requirements and regulatory aspects, a functional specification is defined and the module is designed in the modeling environment, dividing the functions specified in the respective models. The stage finishes with the definition of the unit testing for module validation.

SimDeveloper leverages Simulink to implement and do the unit testing of each module. The module algorithms and models are developed in a simple and effective way, and then the unit testing is run to validate the components.

Once the component has been validated in the prototyping environment, SimDeveloper will automatically generate simulation models in the form of libraries that can be integrated with the rest of the simulation running on Simware platform.



## The Code is Over



Leverage Simware Platform to deploy your simulation assets into any kind of simulation solution, in compliance with the main standards.

The generation process of the module code is that by which the module developed in the SimDeveloper environment is transformed in a software component that can be integrated into a specific target platform.

For this purpose, SimDeveloper generates, in a fast and efficient way, C ++ code of the module, in fully compliant with the DDS/HLA middleware and runtime simulation infrastructure provided by Simware CORE.

### Using Ready-to-Use repositories in SimDeveloper

Since June 2015 we have expanded SimDeveloper with **repositories of Virtual Vehicles and Systems**, included manned, unmanned and weapon assets. The use of this repositories will speed up your development times, reducing your technical risks at the same time.

This repositories are available as **add-ons to SimDeveloper**.

### Software Requisites

To fully utilize the functionality of SimDeveloper the following products are also needed.

- Matlab, Simulink and Real-Time Workshop Embedded Coder
- Microsoft Word 2003 or later (to manage documentation)
- Visual Studio 2008 (for code compilation in Windows)

### For more information

To learn more about SIMWARE SimDeveloper, contact your NADS representative or NADS Business Partner, or visit:

[www.simware.es](http://www.simware.es)



© Copyright NADS 2015  
Nextel Aerospace Defence & Security S.L.  
Camino Cerro de los Gamos,1  
Edificio Negocenter  
28224 Pozuelo de Alarcón MADRID

All Rights Reserved

NADS, the NADS logo, nads.es, simware.es and SIMWARE are trademarks or registered trademarks of International Nextel Aerospace Defence & Security S.L. in Spain, other countries, or both. If these and other NADS trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate Spain registered or common law trademarks owned by NADS at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries.

Other company, product, or service names may be trademarks or service marks of others.

References in this publication to NADS products or services do not imply that NADS intends to make them available in all countries in which NADS operates.

The information contained in this document is provided for informational purposes only and provided "as is" without warranty of any kind, express or implied. In addition, this information is based on NADS' current product plans and strategy, which are subject to change by NADS without notice. Without limiting the foregoing, all statements regarding NADS future direction or intent are subject to change or withdrawal without notice and represent goals and objectives only. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from NADS (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of NADS software.