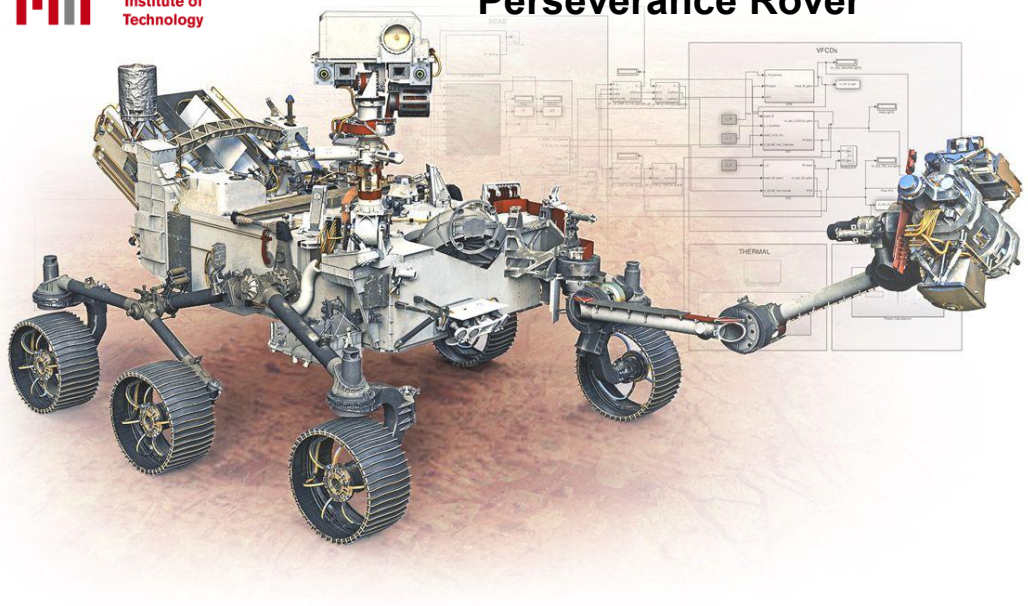
Systems of Systems

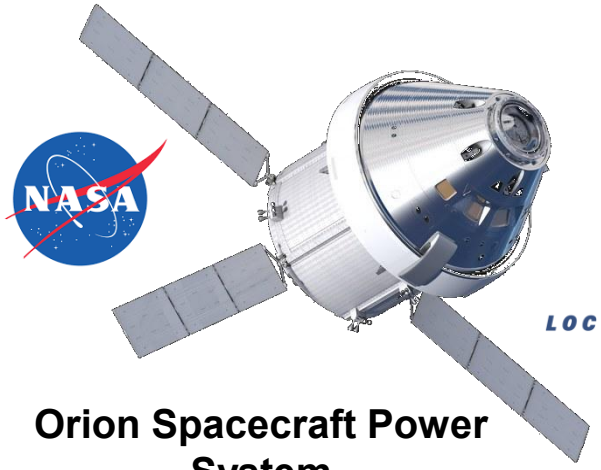
Modelling & Simulation

Massachusetts Institute of Technology

Perseverance Rover



MathWorks



Orion Spacecraft Power System

LOCKHEED MARTIN

F-35 Fleet Performance

And many examples more…

AIRBUS DEFENCE & SPACE

C-295 High-Lift Control Unit – FPGA DAL A

Eurofighter Typhoon Defensive Aid System (DASS)

LEONARDO

Systems of Systems in Defence

Next Generation Weapon System (NGWS) / FCAS

Kaan + MIUS

Global Combat Air Programme (GCAP)

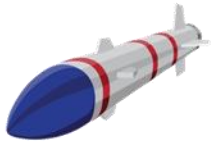
Aeronautics

Eurodrone

Main Ground Combat System (MGCS)

Land

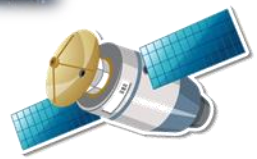
Combat Cloud



Weapons

DragonFire

Naval



Space

EU HYDEF

F110

IRIS²

S-80

Dreadnought

Systems of Systems

Adaptation

Heterogeneous Platforms

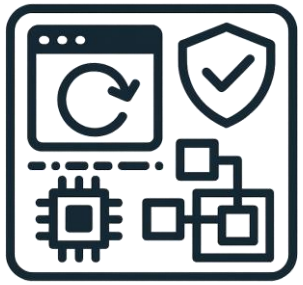
Distributed Intelligence

SYSTEMS OF SYSTEMS

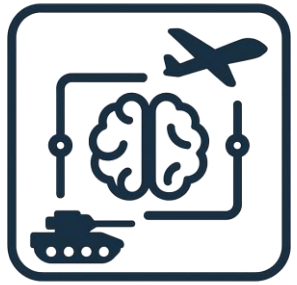
Increased Autonomy

Man-Unmanned Teaming

Real-Time Connectivity



SOFTWARE-DEFINED DEFENCE



AI IN DEFENCE

Fast SW Updates

Software-Defined Assurance

Hardware Abstraction, 1:N

Modular Architectures

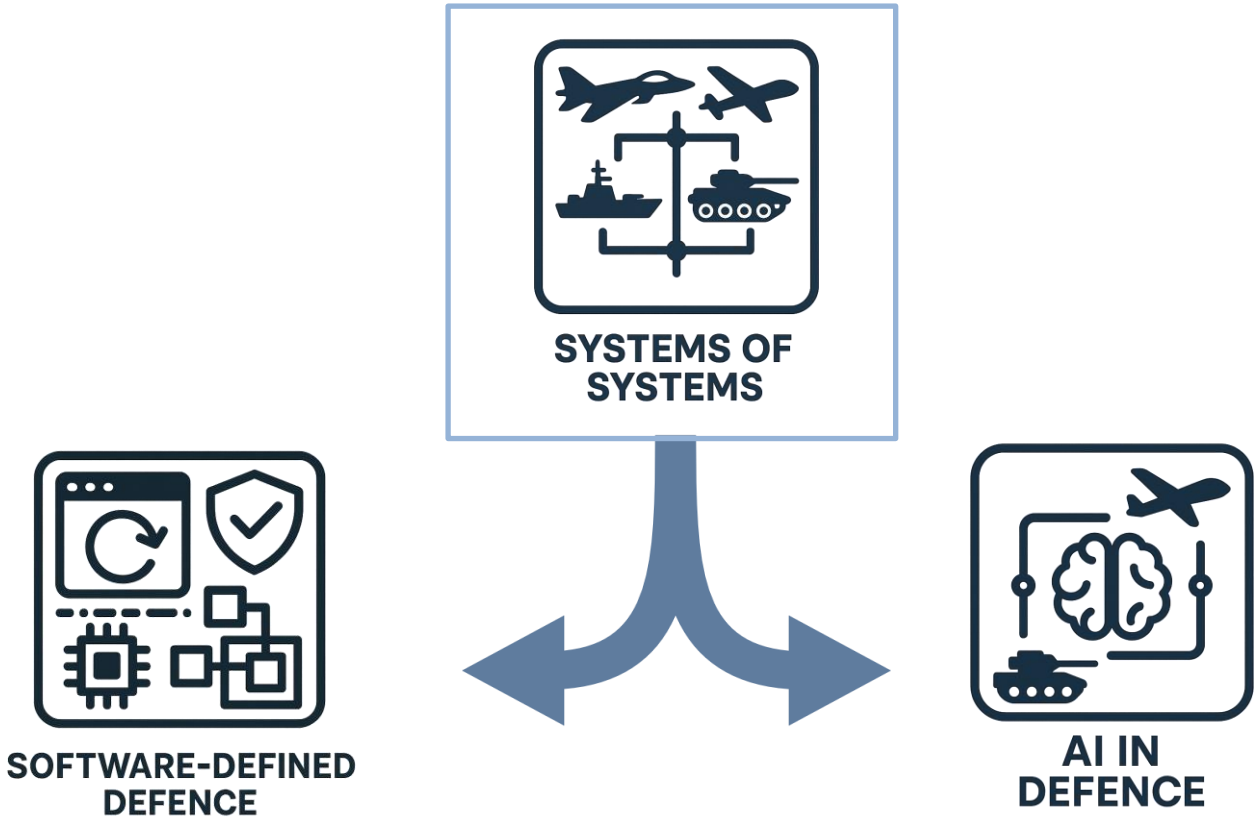
Modelling & Simulation

Massive Data

Data: Real, Augmented, Synthetic

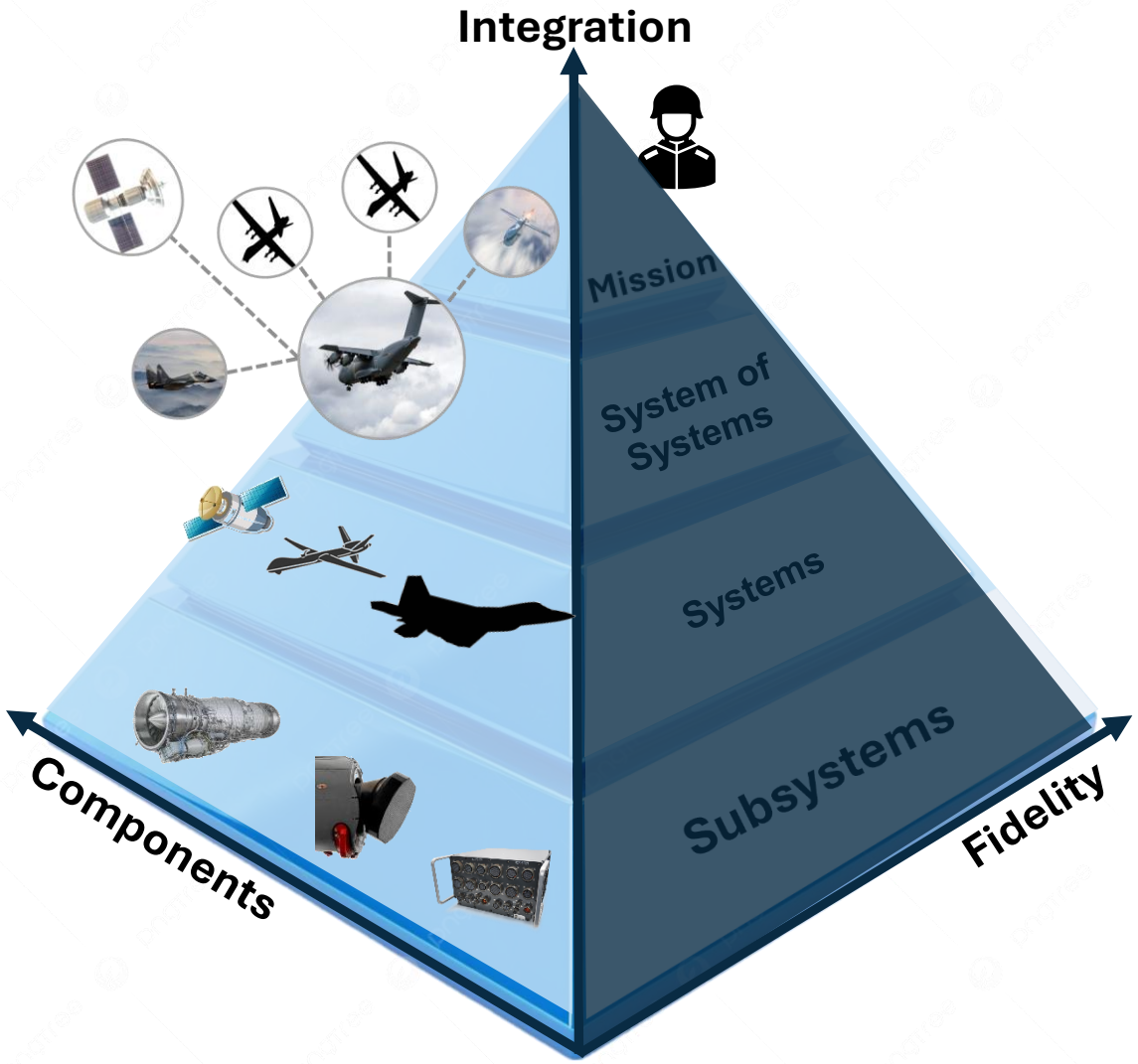
AI Assurance

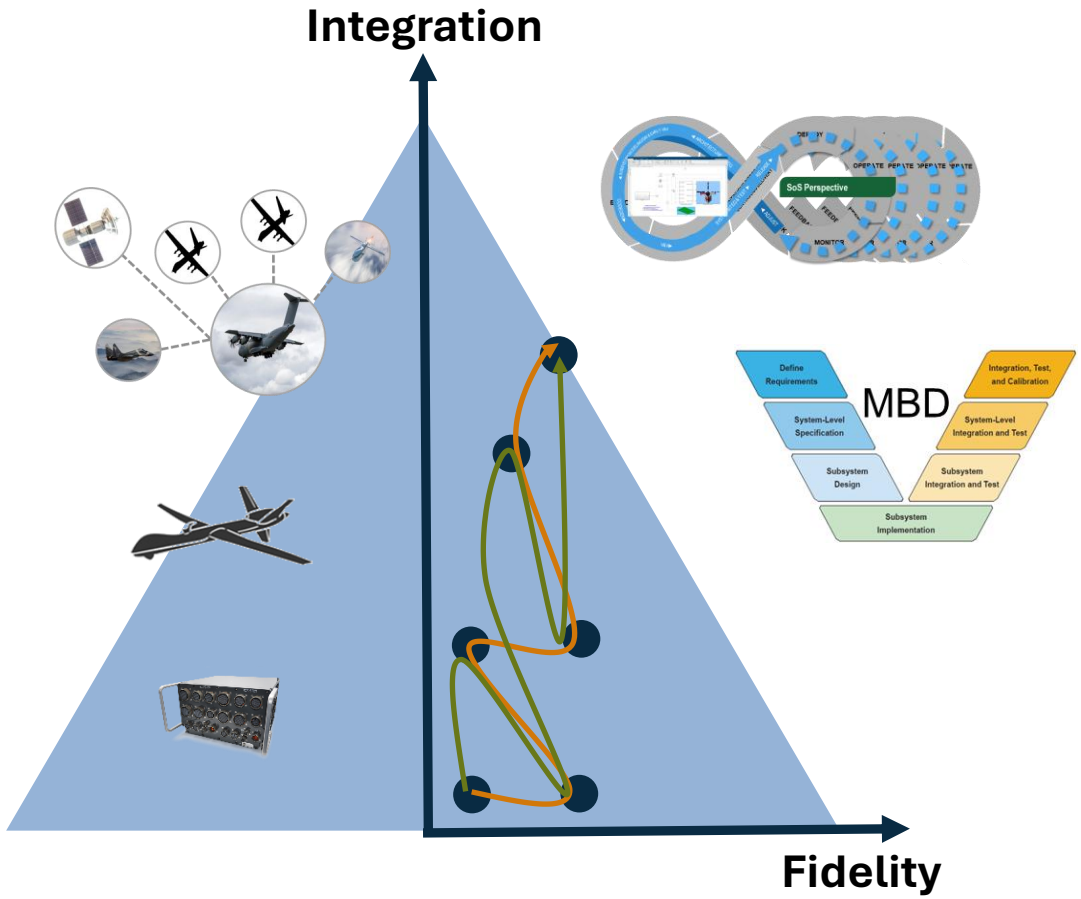
Embedded AI



Modelling & Simulation

Design of Systems of Systems





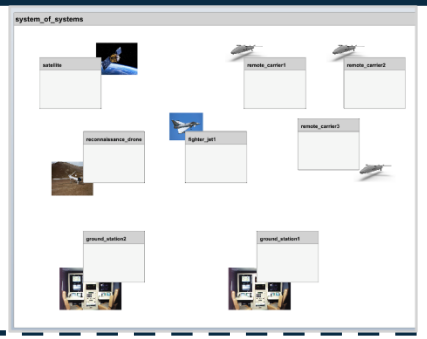
- Different Perspectives
- Different Tasks
- Different Levels
- Numerous Iterations

Communication and Traceability using Models

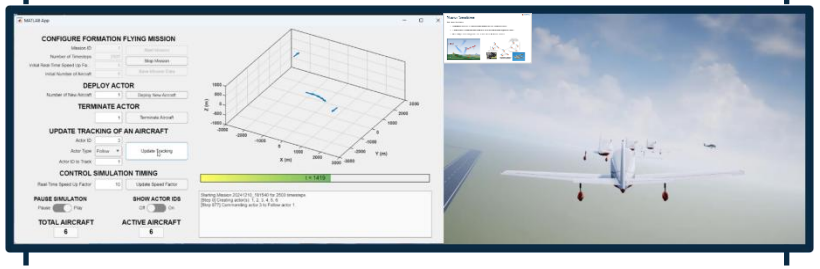
Integration & Fidelity

Integration

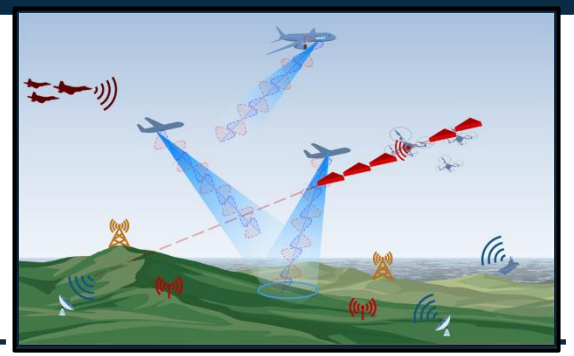
Mission & SoS Architecture



Mission Simulation

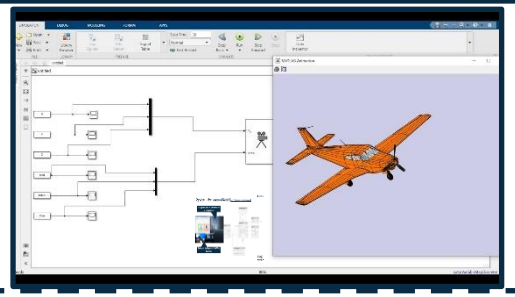


SoS Deployment

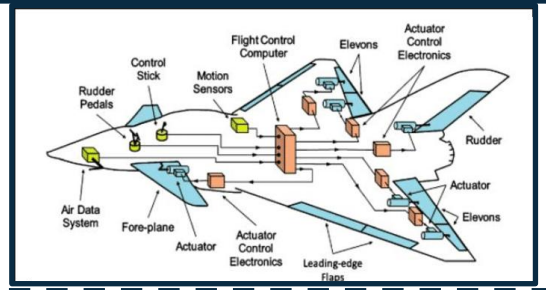


e.g. Aircraft Architecture

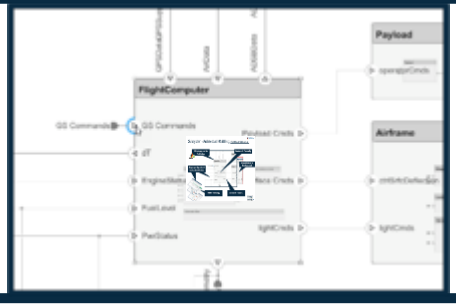
e.g. Aircraft Simulation Model



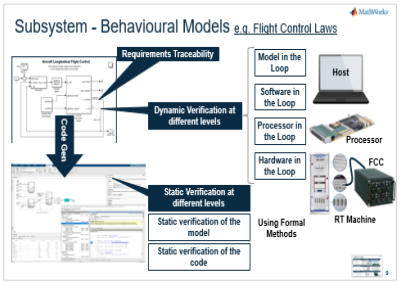
e.g. Aircraft Demonstrator



e.g. Architecture FCC



E.g. Behavioural Flight Controls



e.g. Flight Control Computer

Fidelity

Subsystem - Architectural Model e.g. Flight Control Computer

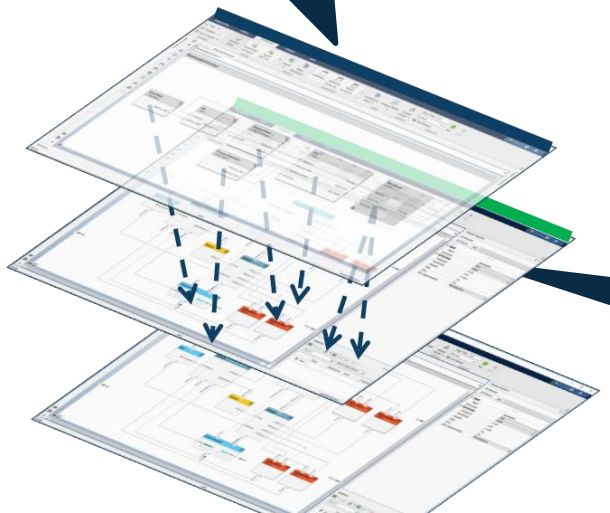
Static Analysis of the Architecture

Scripting

Requirements Traceability

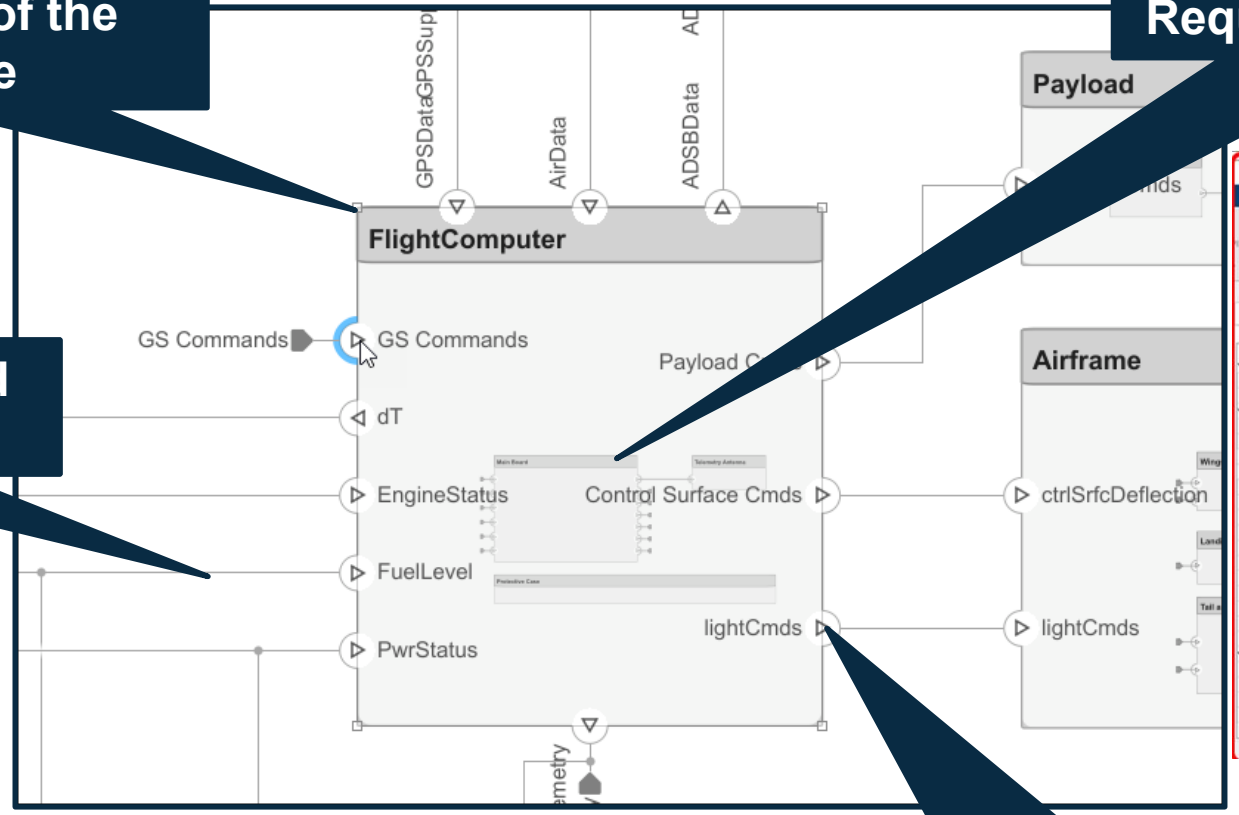
Attribute definition for the different components

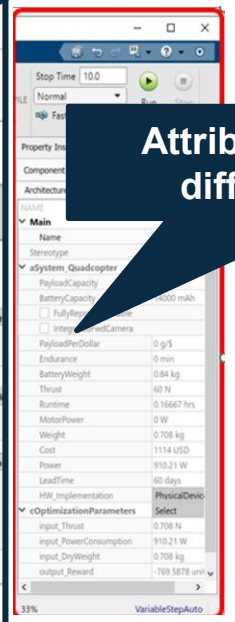
Mapping of physical and functional architectures



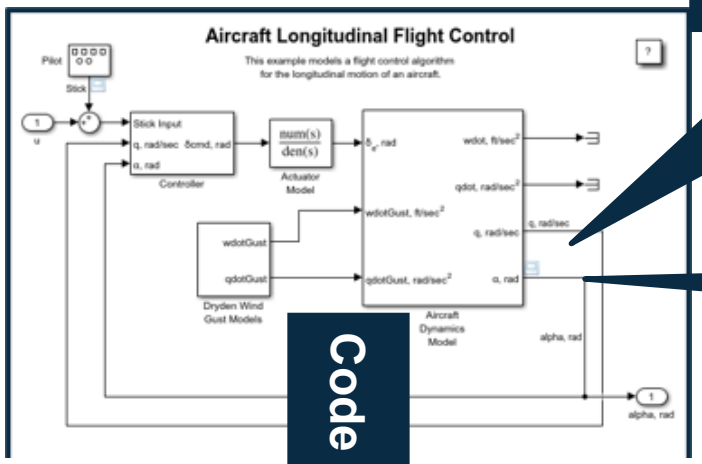
HW/SW Partitioning

Interfaces & Protocols





Subsystem - Behavioural Models e.g. Flight Control Laws

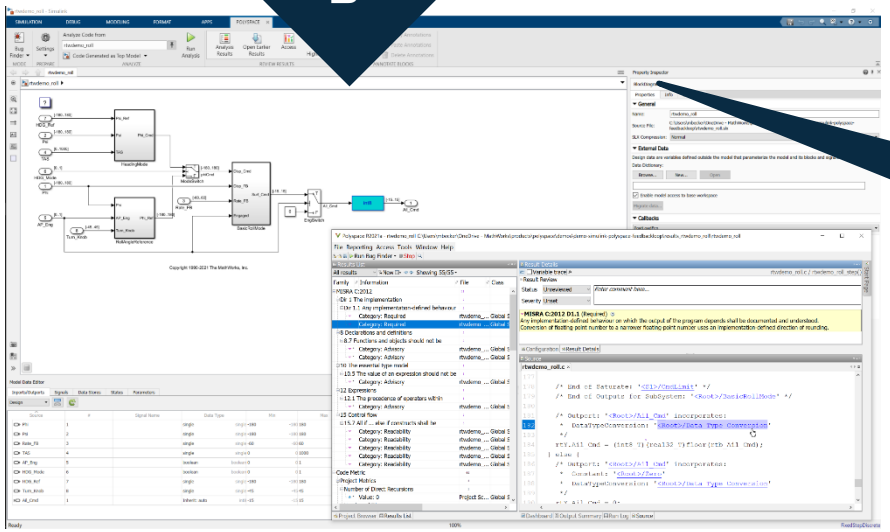


Requirements Traceability

Dynamic Verification at different levels

- **Model in the Loop** — Host
- **Software in the Loop**
- **Processor in the Loop** — Processor
- **Hardware in the Loop** — FCC, RT Machine

Code Gen



Static Verification at different levels

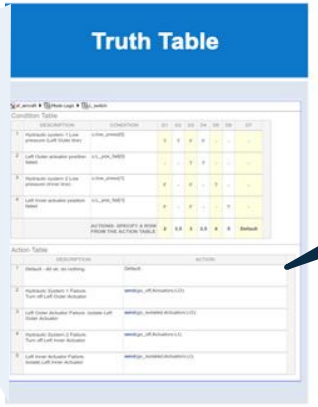
- **Static verification of the model**
- **Static verification of the code**

Using Formal Methods

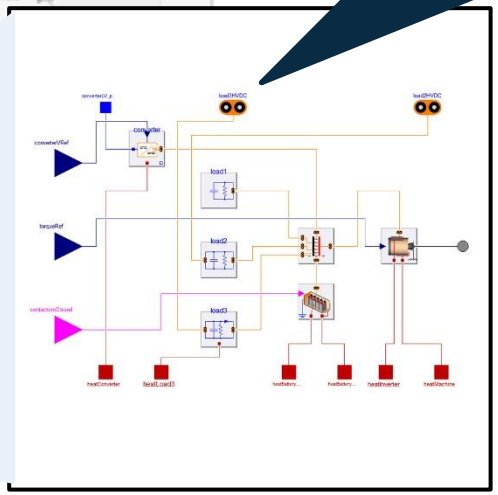
System – Architectural Model e.g. Drone

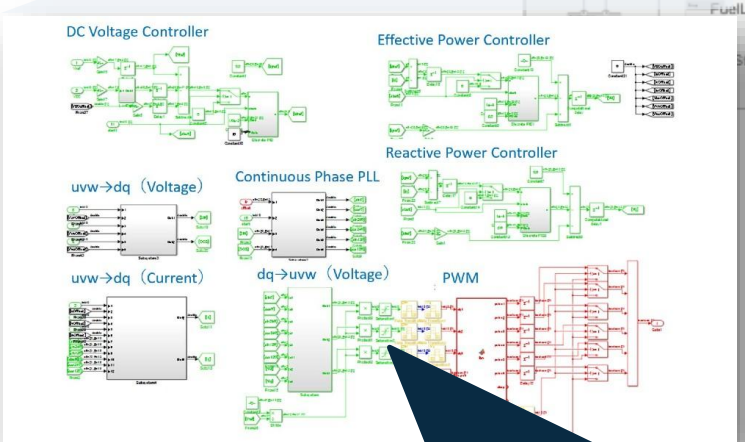
High Fidelity: FCC

Low Fidelity: state machines, tables, etc.



Medium Fidelity: models using 3rd party tools or code. E.g. FMI, HDL, C, C++, etc.

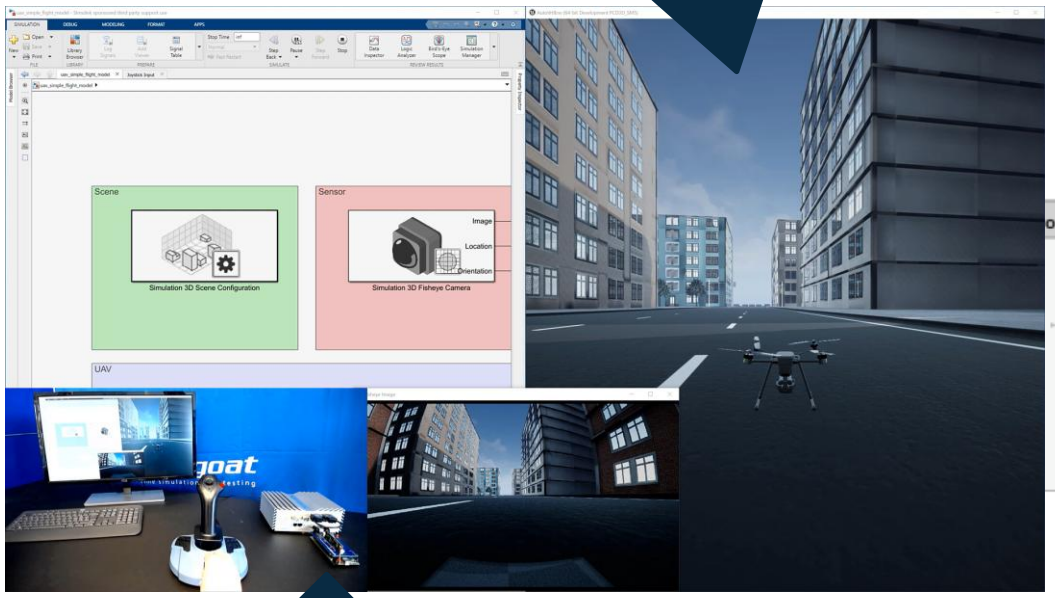


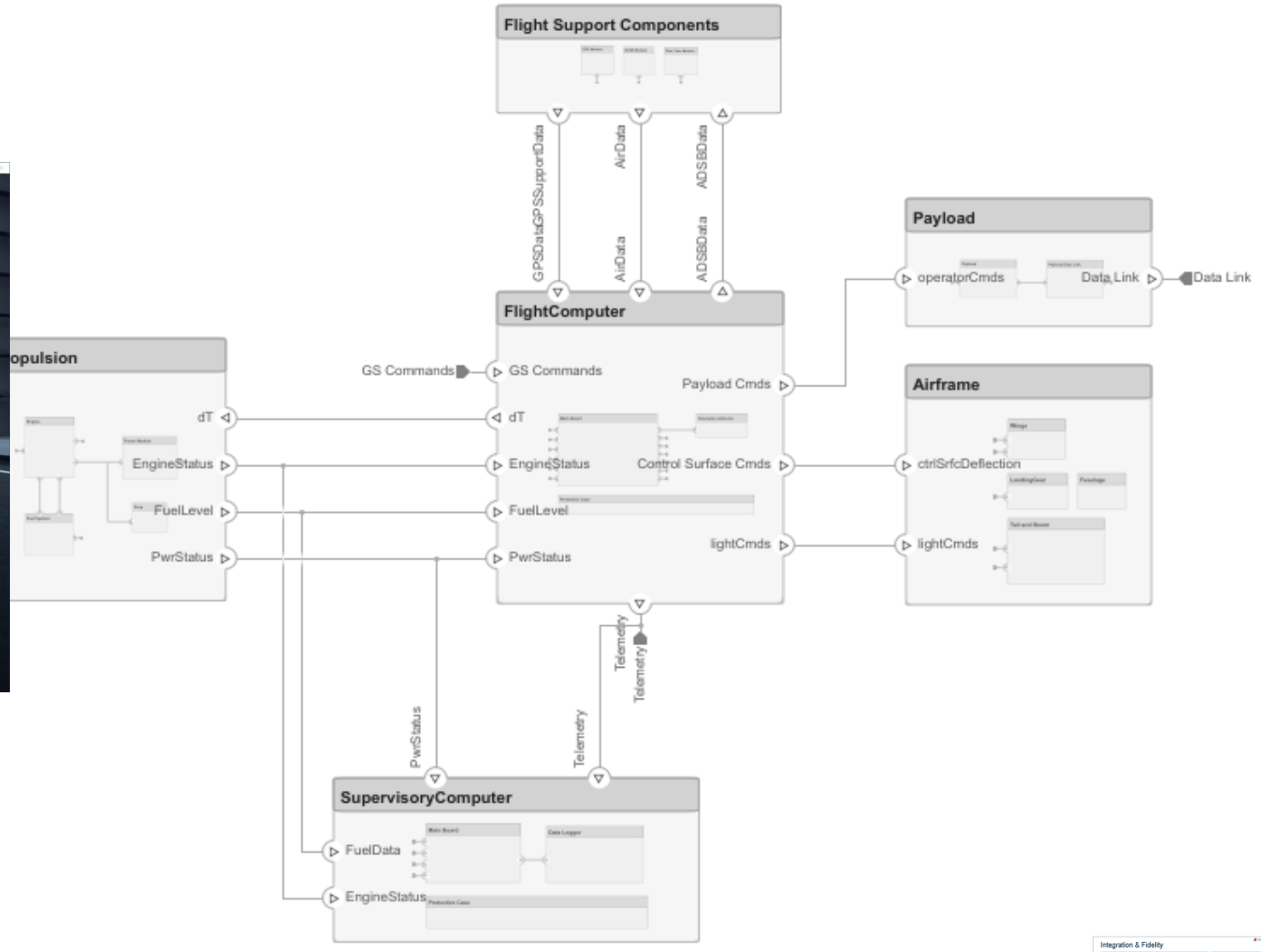


Medium Fidelity: Simulink, Simscape

System – Behavioural Model e.g. Drone + environment

Integration with visualization and 3D environments



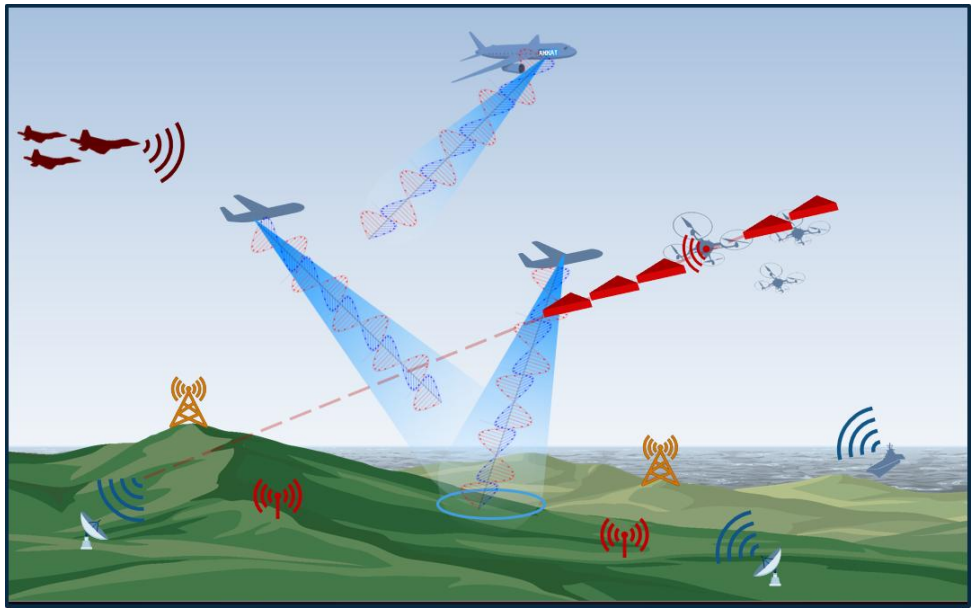


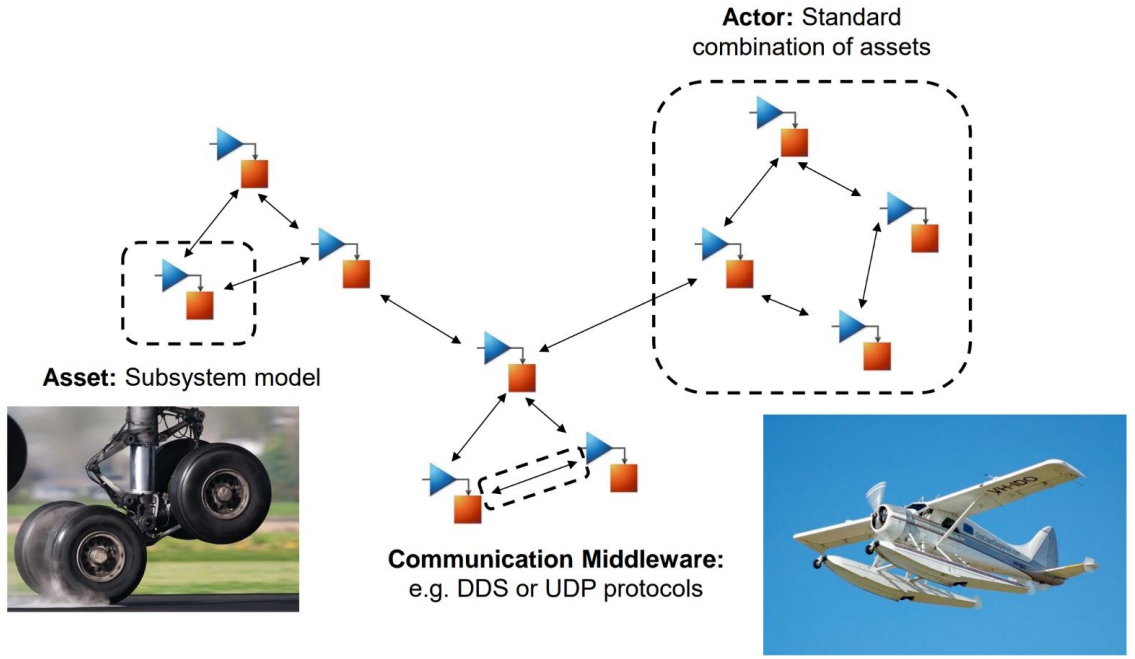
Human-in-the-Loop and Real-Time Machines

Mission Simulation

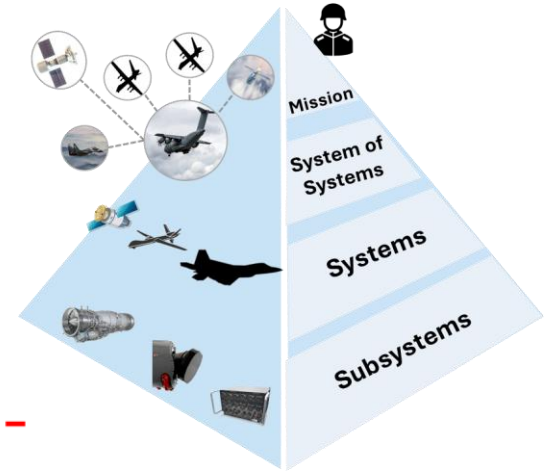
Distributed Simulation:

- **Scalability**: possibility of distributing simulations across different machines.
- **Collaboration**: modular architecture to favour collaboration in heterogeneous teams.
- **Reusability**: actors ready to be used in simulations of different missions.





Mission Simulation: when?



Complex & time-consuming mission definition

Whole scenario simulation

Computational Limit Upper Bound

Complex and time-consuming actor definition

Real-time vehicle training simulator

Comms. network overhead exceeds benefits

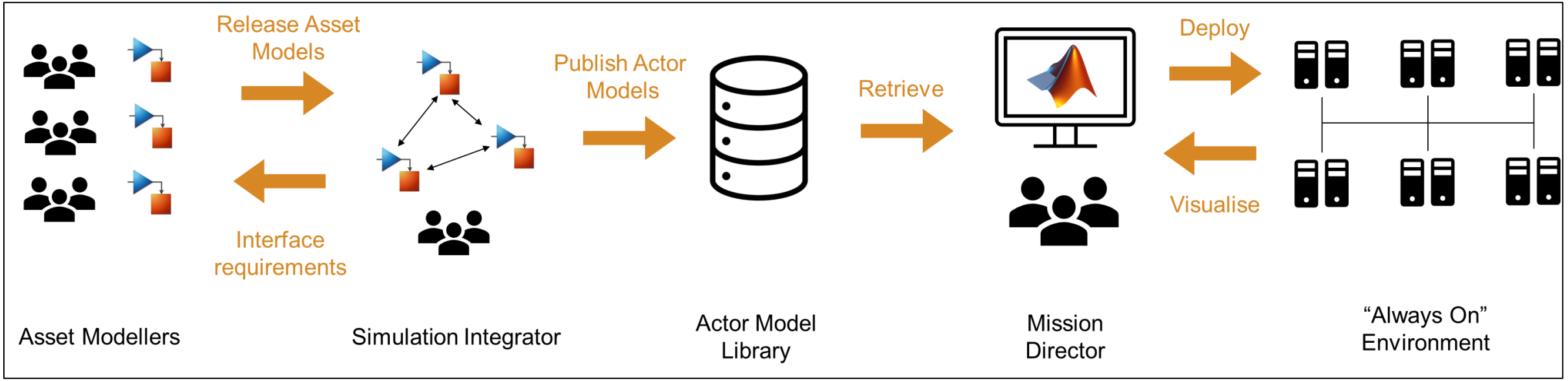
Detailed software architecture

Number of Actors (Increasing mission scale)

Number of Assets per Actor (Increasing mission fidelity)

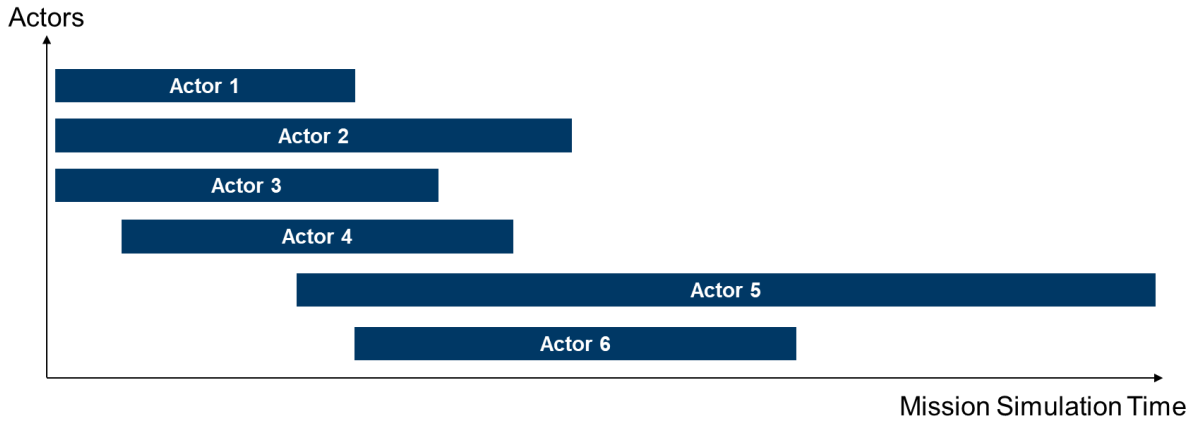
Mission Simulation

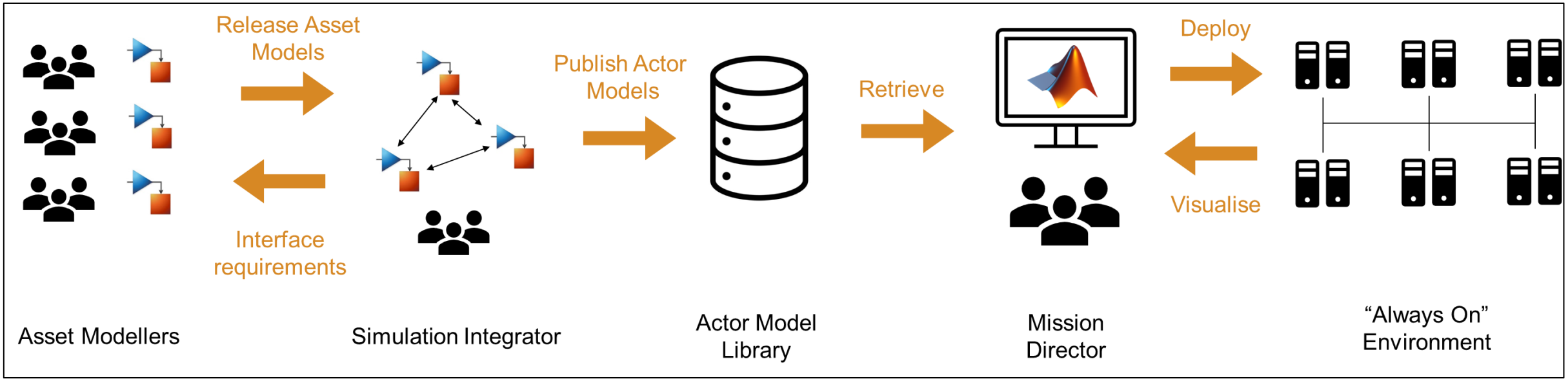
- Full workflow, from asset modellers to mission director.
- *“Always-on Environment”.*
- Create model libraries with reusable actors to be used in different missions.

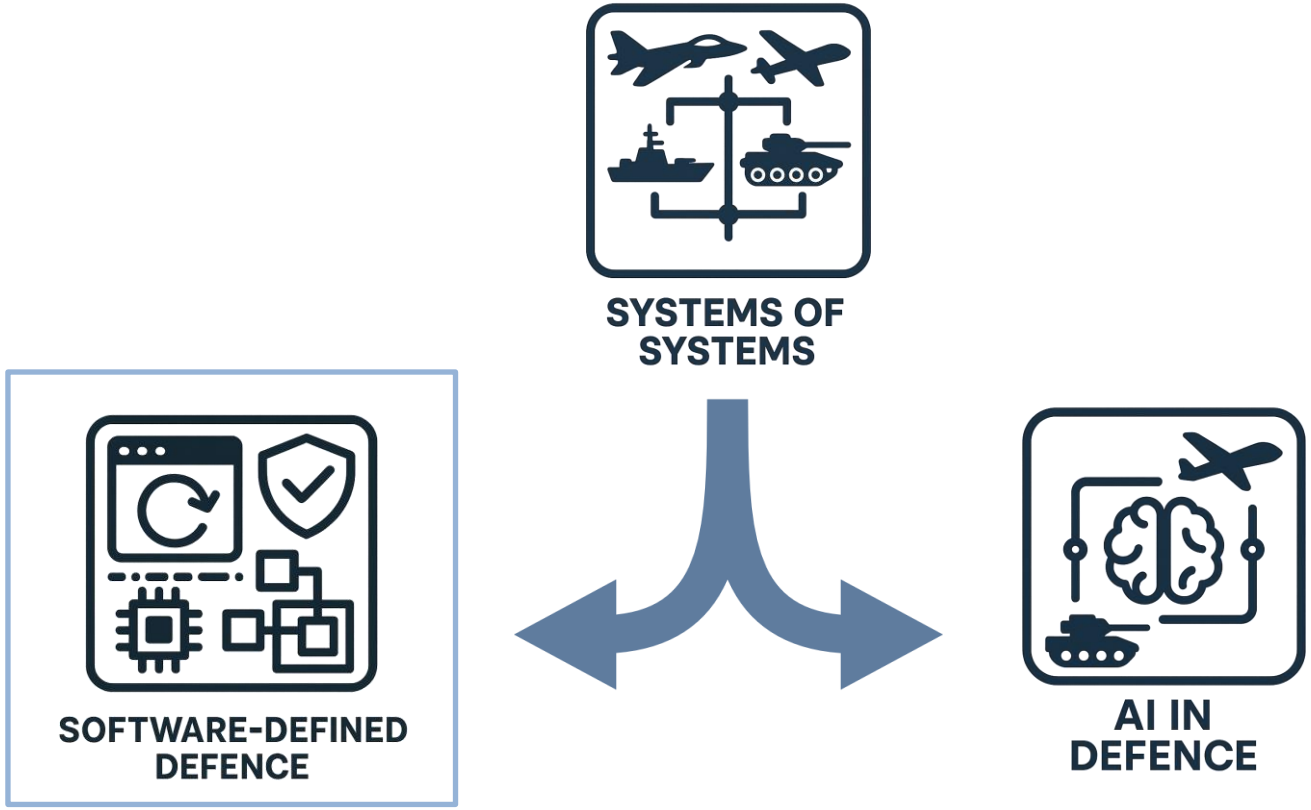


Mission Simulation

- **Architecture**: Which actors are present in the mission, and how do they interact with each other?
- **Timeline**: How many actors are there, and when are they deployed, concluded, or re-parameterized?







Modelling & Simulation

Example: update lifecycle

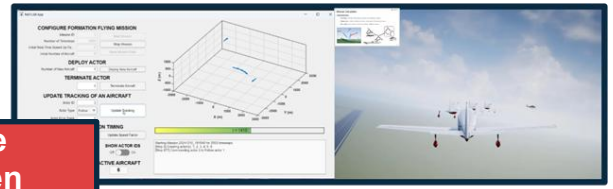
Enhancement test in mission simulation environment

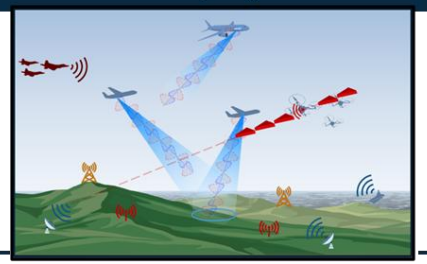
Identified enhancement

Systems of Systems Architecture

Distributed Simulation

SoS Deployment



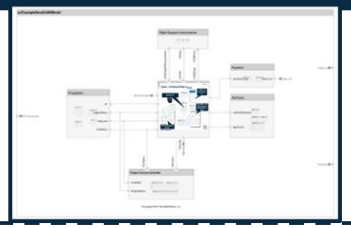


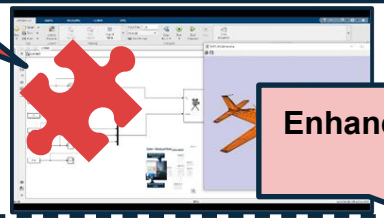
Enhancement tested in extreme situation and configurations even outside of misión environment

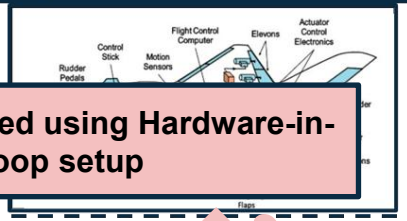
e.g. Aircraft Architectu...

e.g. Aircraft Simulation Model

e.g. Aircraft Demonstrator





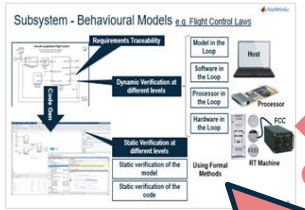


Enhancement tested using Hardware-in-the-Loop setup

e.g. Architecture FCC

E.g. Behavioural Flight Controls

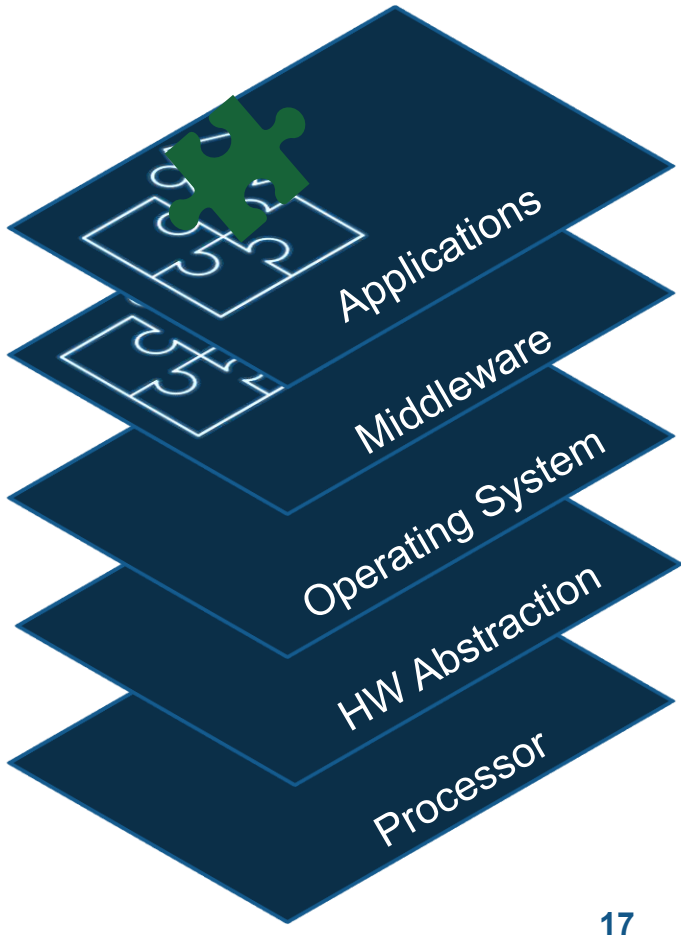
e.g. Flight C... Computer



Integration

Fidelity

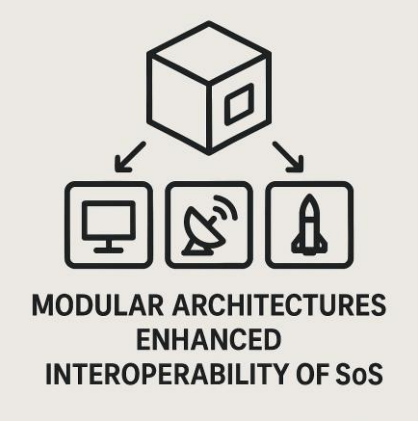
Enhancement exhaustively tested in HW or SW implementation



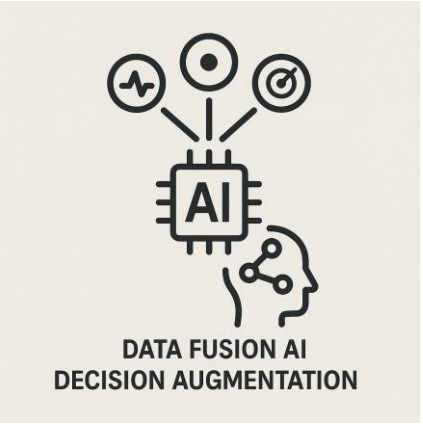
What is Software-Defined Defence (SDD)?

HW Abstraction – 1: N
Decoupling software from hardware development. Enhancing modularity & upgradeability

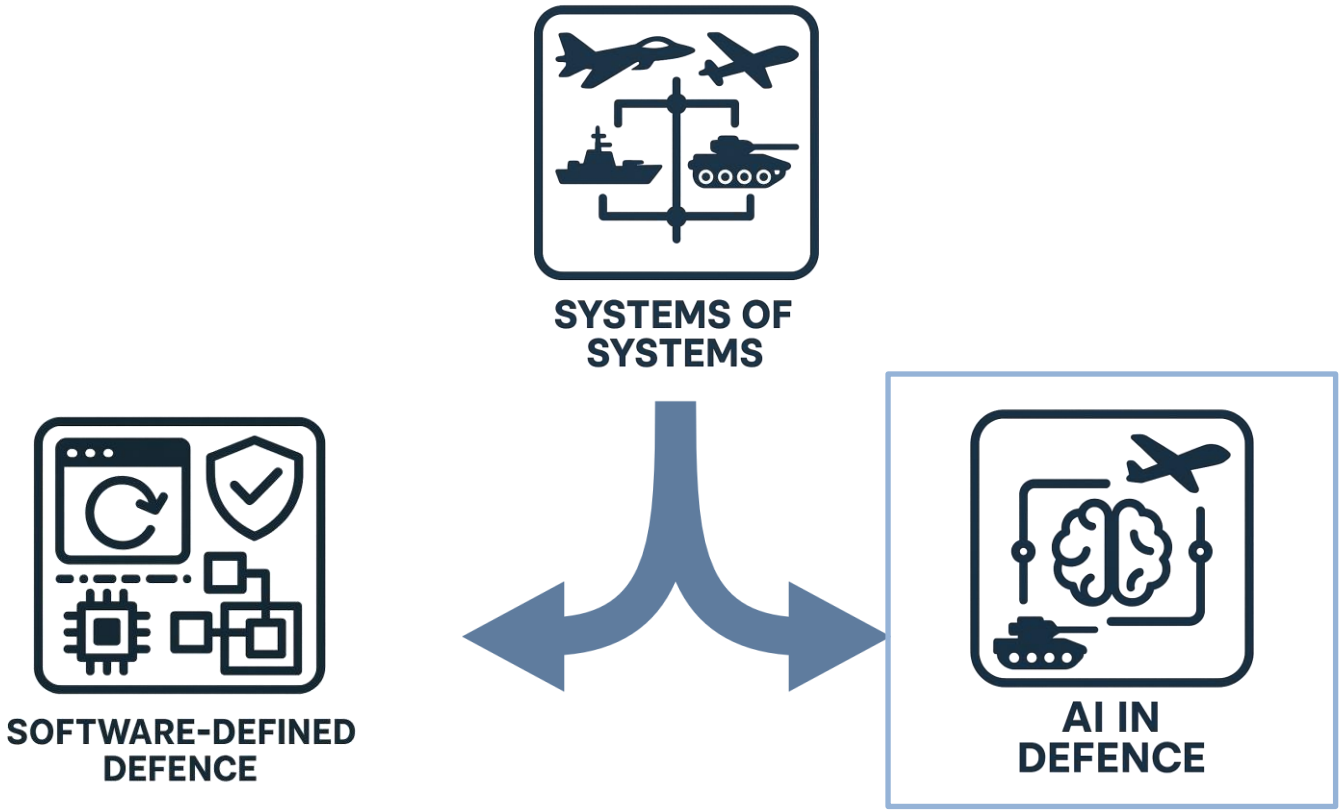
Rapid field updates
Continuous capability enhancement – SD Assurance Software-centric systems can be patched, updated, or reconfigured in the field, enabling new capabilities, vulnerability patches, or adaptation to emerging threats without full hardware replacement.



Enhanced interoperability and allow system-of-systems integration bringing open, standardized interfaces and **modular architectures** so that heterogeneous platforms can interoperate, share data, and be orchestrated as systems.



AI, data fusion, and decision augmentation
The flexibility of software-defined platforms allows embedding of AI, data fusion, and real-time analytics.

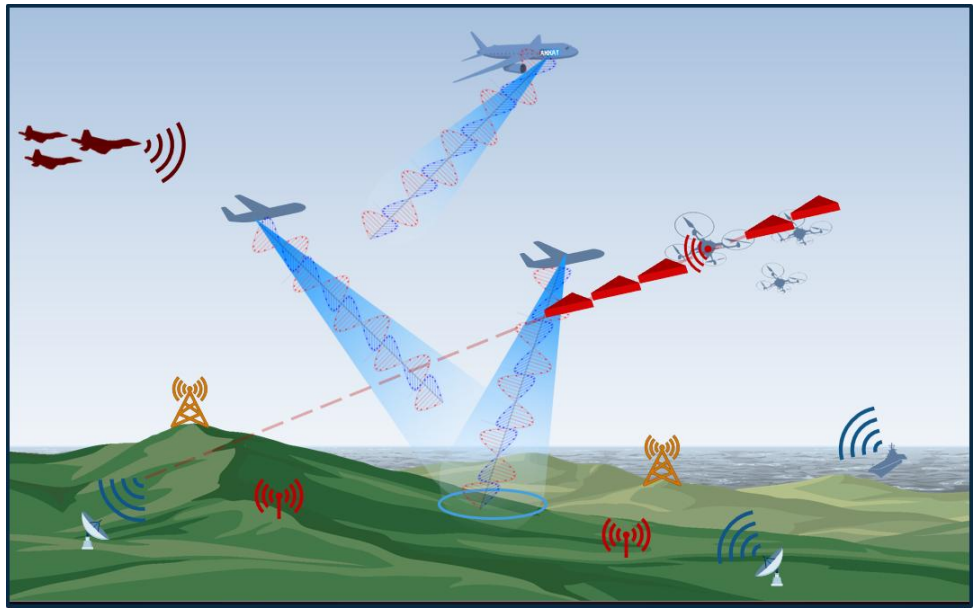


Modelling & Simulation

AI in Aerospace and Defence

AI as a design tool

AI as part of the systems logic

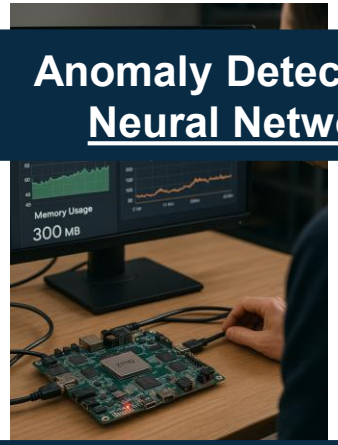


- **Acceleration of simulations**, e.g. Deep Learning for Reduced-Order Models (ROM).
- **Generative AI**, e.g. assistance with model, code, and/or test vector generation. Co-pilots.
- **Predictive maintenance**, e.g. estimating Remaining Useful Life (RUL) of engines using deep Convolutional Neural Networks (CNNs).

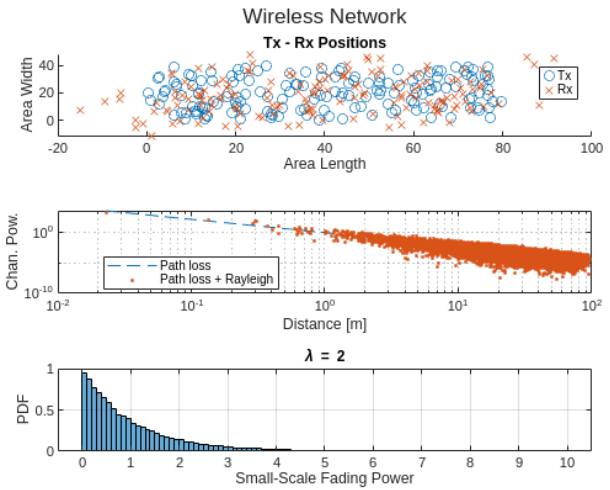
Data Preparation	AI Modeling	Simulation & Test	Deployment
Data cleansing and preparation	Model design and tuning	Integration with complex systems	Embedded devices
Human insight	Hardware accelerated training	System simulation	Enterprise systems
Simulation-generated data	Interoperability	System verification and validation	Edge, cloud, desktop

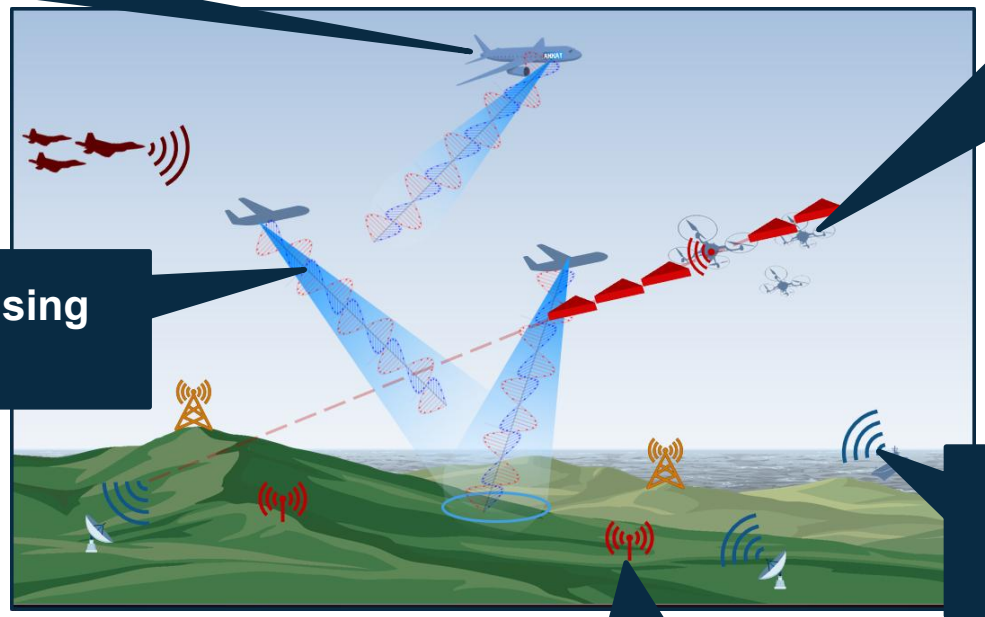
AI-Enabled Systems

Anomaly Detection using Recurrent Neural Networks in FPGA Link



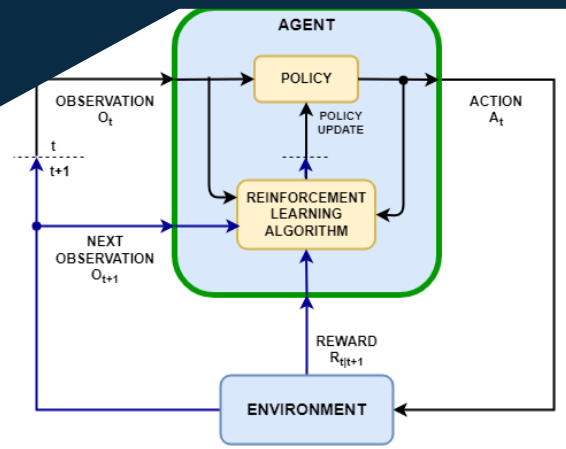
Wireless Resource Allocation Using Graph Neural Network Link



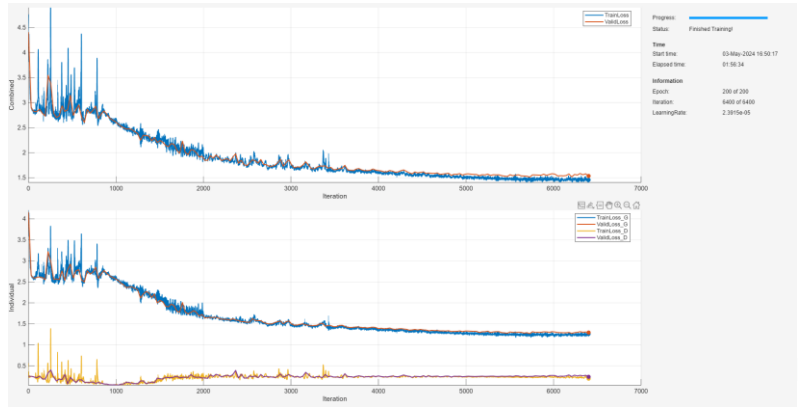


Use of LLMs to decode voice commands

Reinforcement Leaning Agents for autonomous systems Link

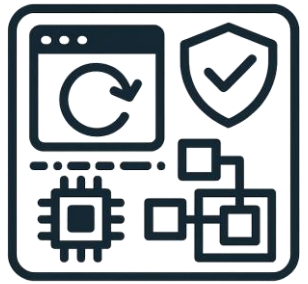


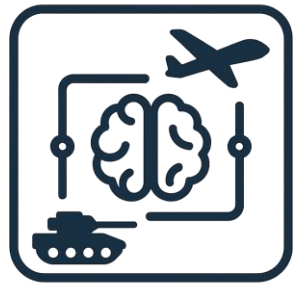
Denoise Signals with Generative Adversarial Networks Link



Conclusion

- **Modeling and simulation** are crucial for the design of **Systems of Systems**.
- **Simulate everything** — from missions to embedded system implementations.
- **Full traceability** between design phases and teams is achievable.
- Concepts such as **abstraction**, rapid **updates**, and **modular** architectures are **inherent to Model-Based Design.**
- **AI is trending — but we can make it real**: compression, interaction with other tools, and more.





Thank You

Dr.-Ing. Juan Valverde *jvalverd@mathworks.com*

MathWorks®
Accelerating the pace of engineering and science