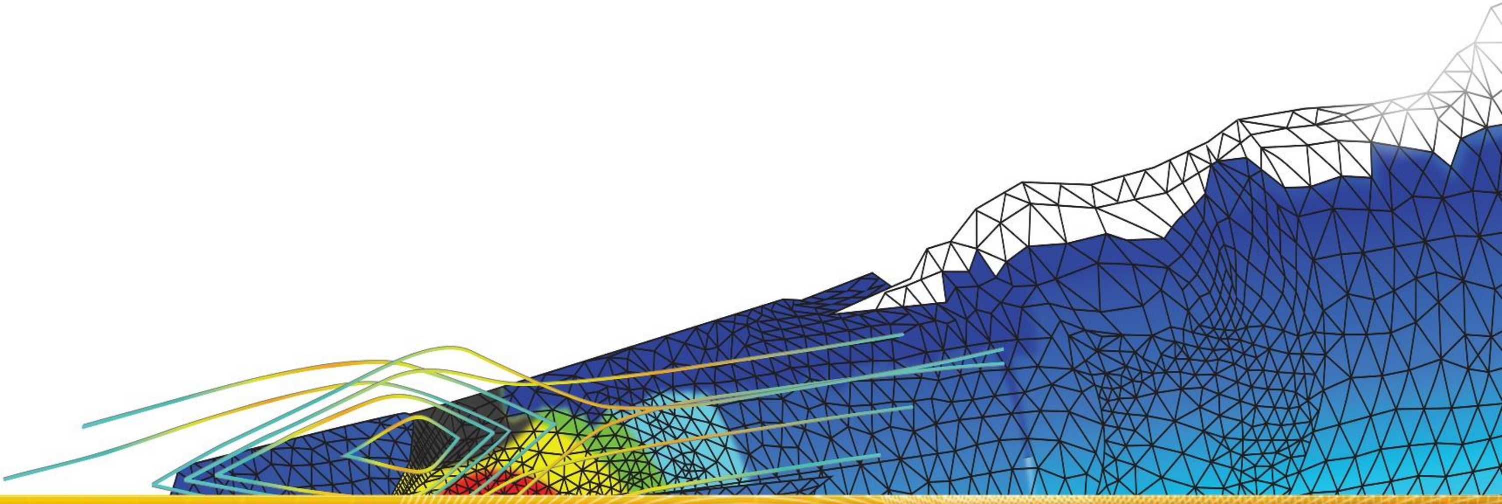
Realize Your Product Promise®

ANSYS®

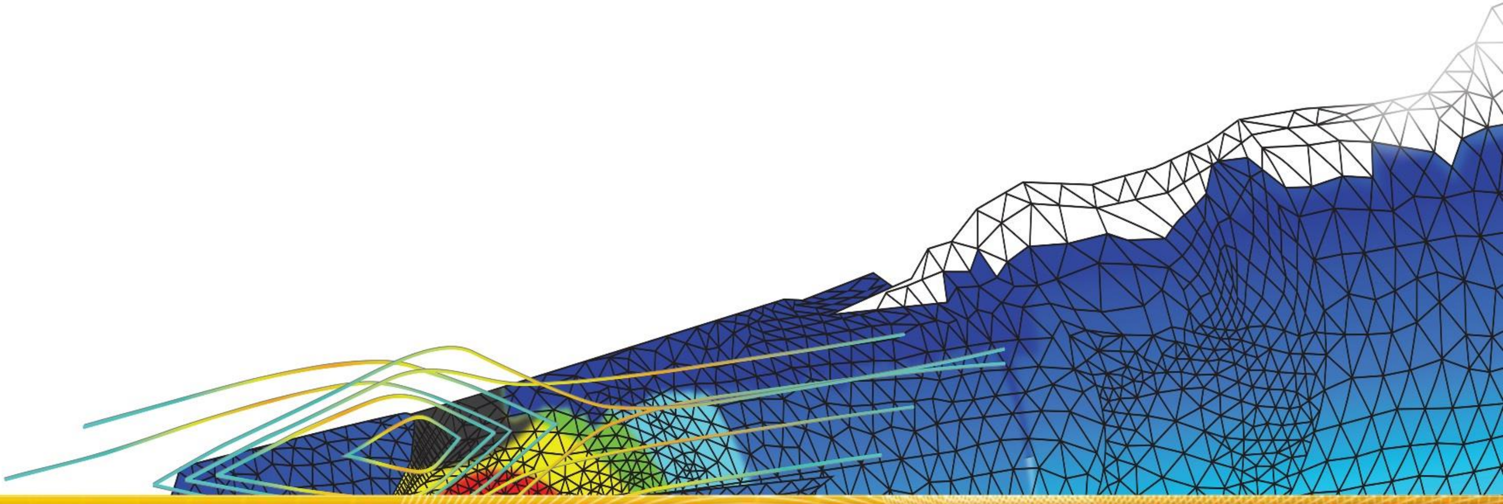
What is Simplorer



Agenda

- Systems Overview
- Systems Challenges and ANSYS Solutions
- Simplorer
- Successful Stories

Systems Overview



Systems – What We Mean

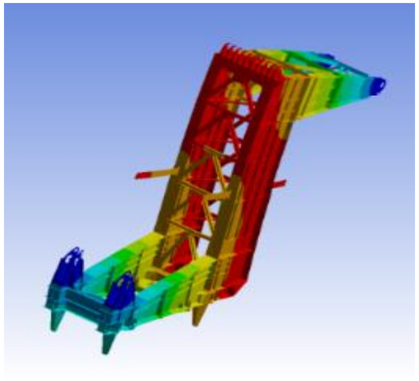
- **System** =
  - Well-defined part of an Industrial Product or Asset: Aircraft/Vehicle/Power Plant/Oil Rig/Pipeline/Train, etc….
  - Delivers a particular functionality

  - When the product is *simple* (a pump, a battery), the product itself is a system
  - When the product is very complex (a plane, a car…), it is generally a collection of systems, with sub-systems

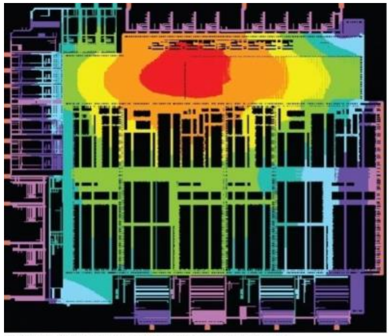
- **Systems of Systems** refers to the interaction of many systems in a distributed manner: an air traffic control, an electrical grid, a subway covering a city… we don't operate at this level

Systems are made of

- **Their Body**: Physical components that sustain multiple physical phenomena: stress, vibration, fluid/structure interactions, electromagnetic interference, noise, flutter, fatigue, heating, icing, combustion, deformations, aging, radiation, etc…

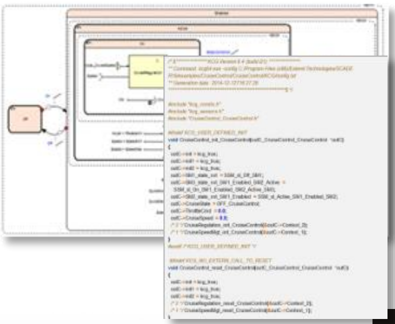


- **Their Nervous system**: Electronic components: sensors, actuators, integrated circuits, chip/package/electronic system, network interfaces

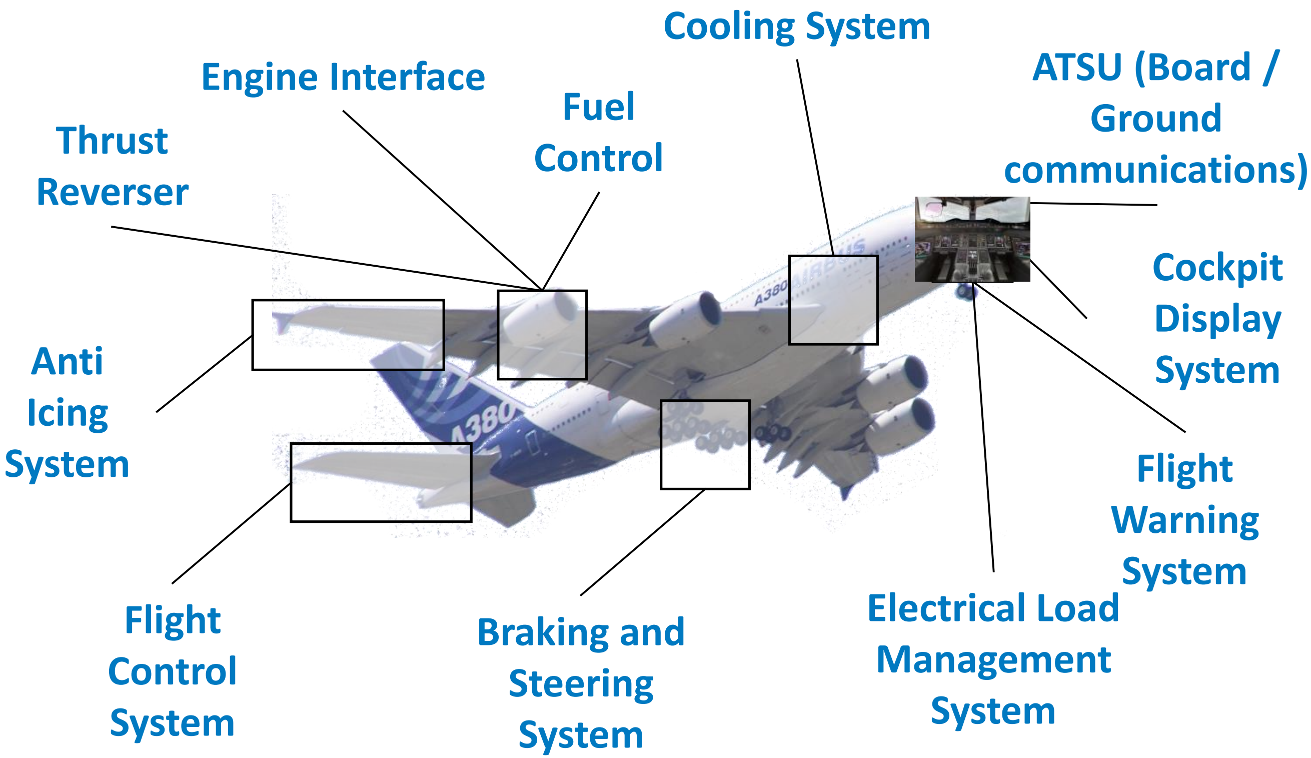


RedHawak - Chip-level power integrity - Dynamic voltage drop map

- **Their Intelligence**: Software components: controls, user interfaces, databases, data management, maintenance and health monitoring



Example: A&D



Cooling System

Engine Interface

Fuel Control

ATSU (Board / Ground communications)

Thrust Reverser

Cockpit Display System

Anti Icing System

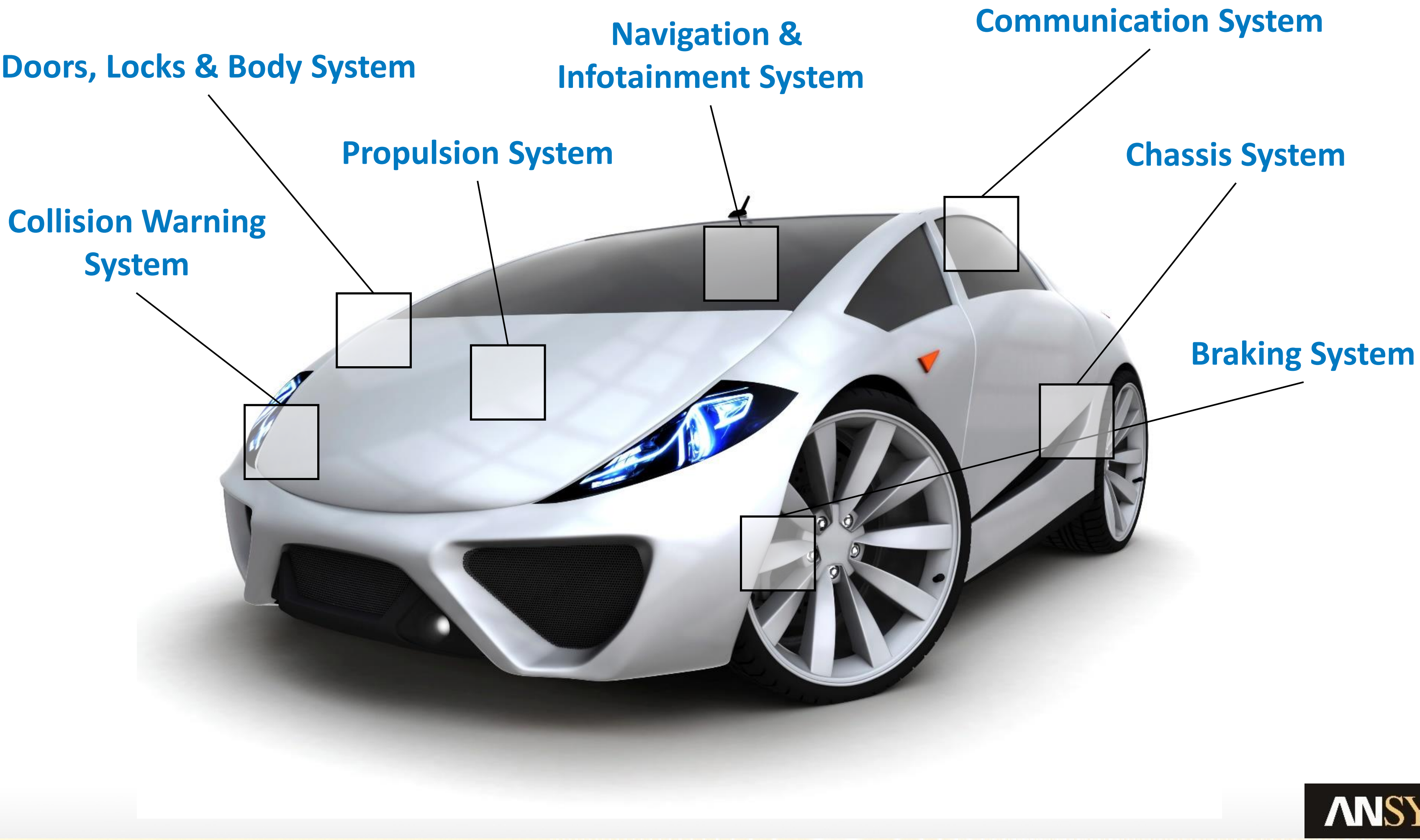
Flight Warning System

Flight Control System

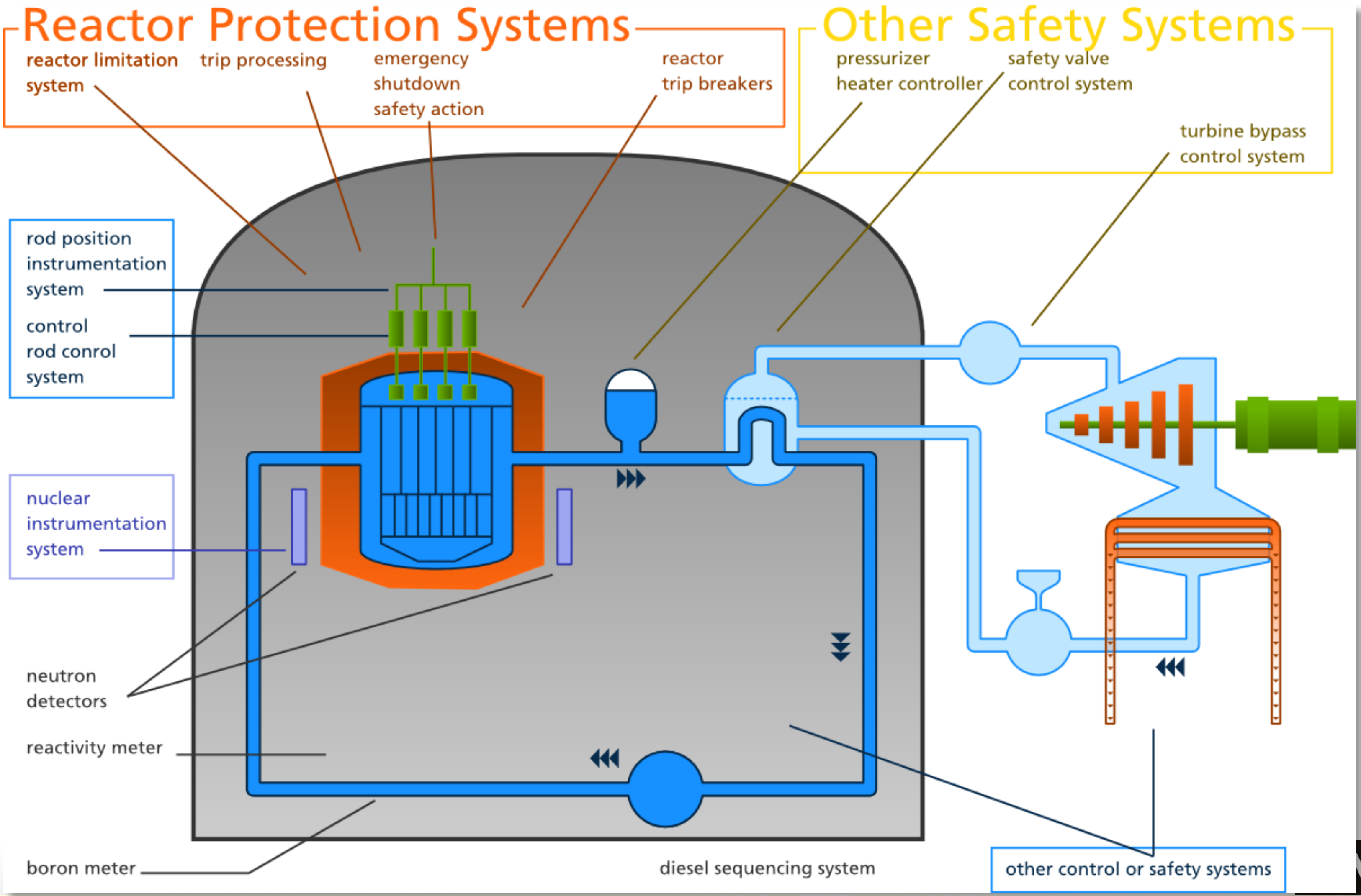
Braking and Steering System

Electrical Load Management System

Example: Automotive



Example: Nuclear


Reactor Protection Systems

reactor limitation system

trip processing

emergency shutdown safety action

reactor trip breakers

Other Safety Systems

pressurizer heater controller

safety valve control system

turbine bypass control system

rod position instrumentation system

control rod conrol system

nuclear instrumentation system

neutron detectors

reactivity meter

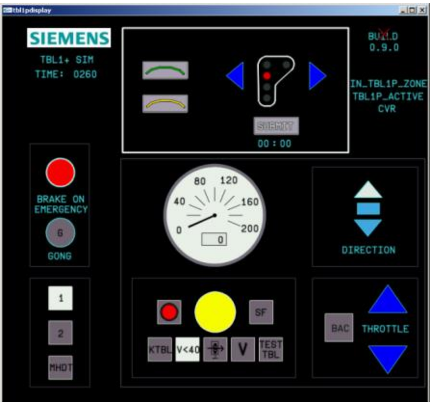
boron meter

diesel sequencing system

other control or safety systems

Example: Railways

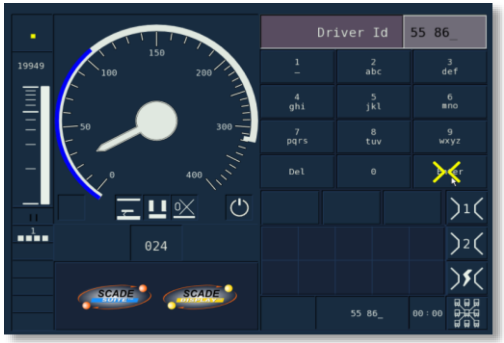
Traffic Tables:
CONTROL CENTER



Simulators

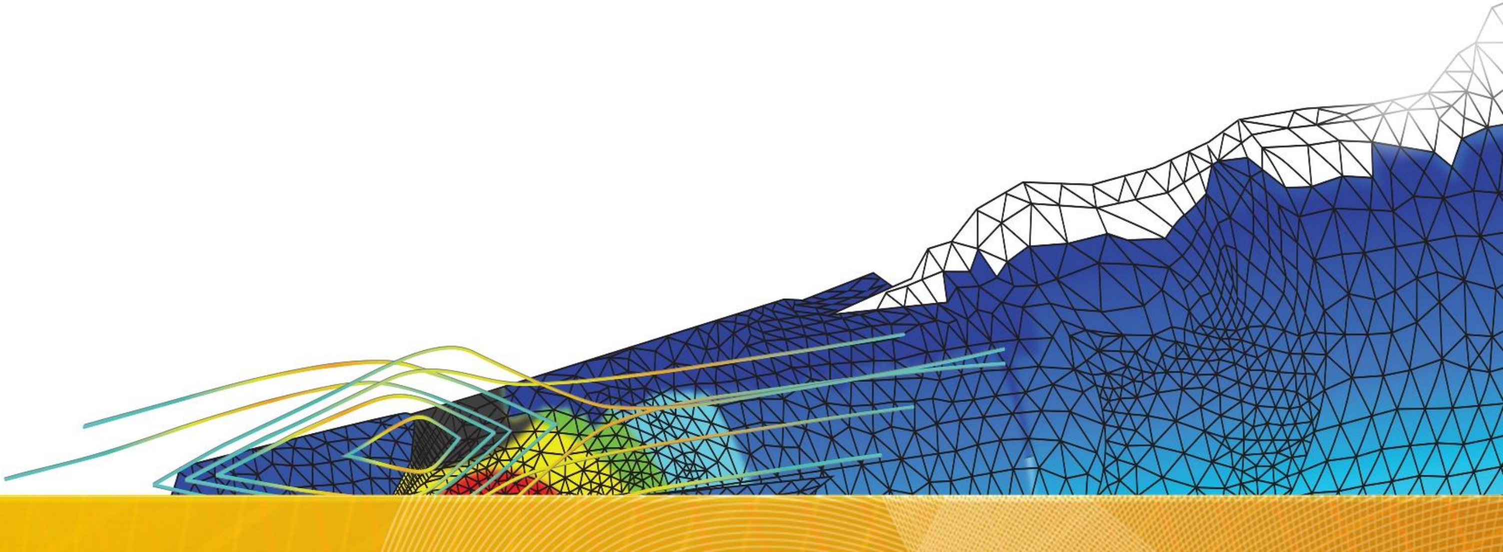
Routes & Safeguards:
INTERLOCKING

Performance & Safety:
TRAIN CONTROL SYSTEM

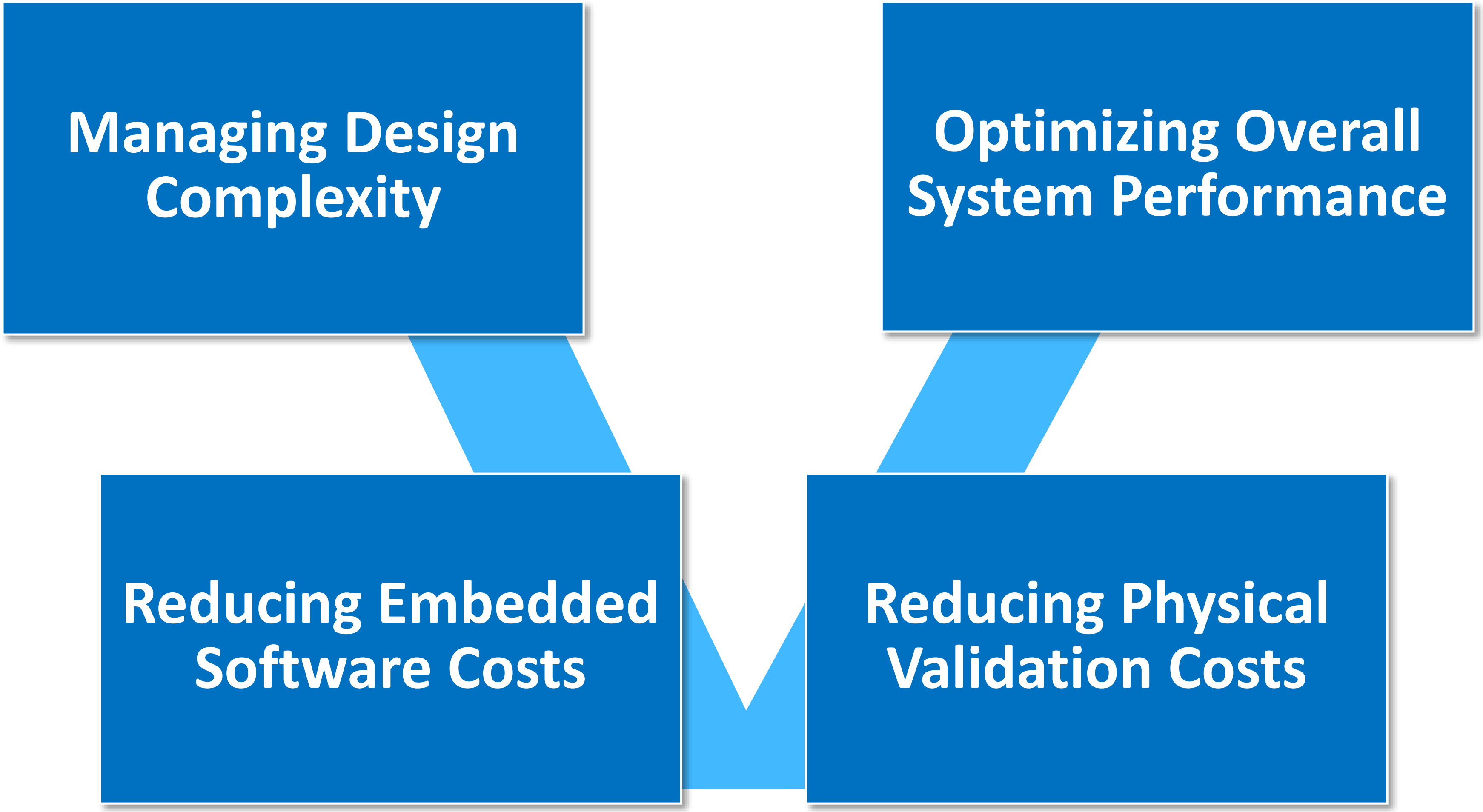


Driver Machine Interfaces

Systems Challenges and ANSYS Solutions



Systems Development Challenges



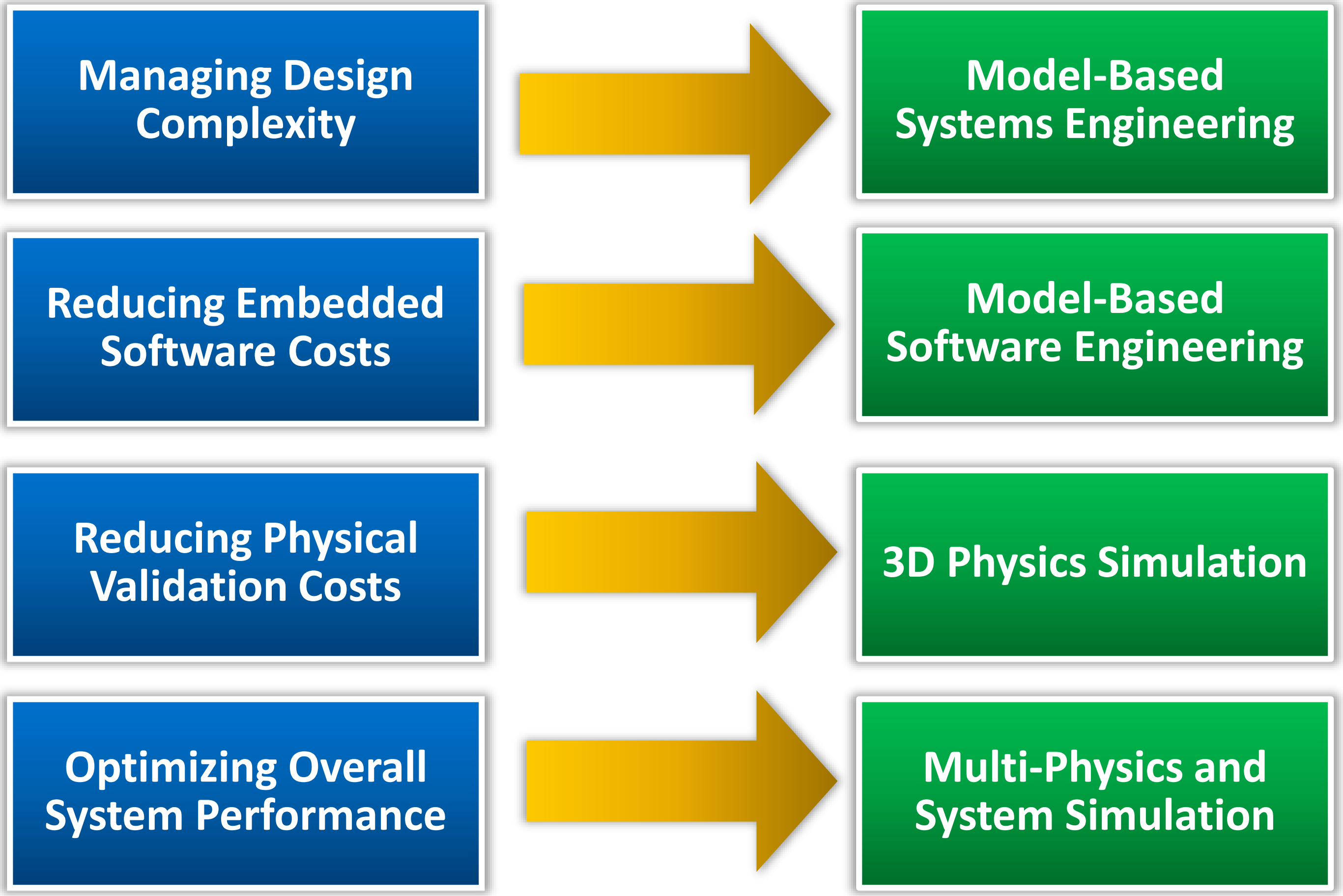
Managing Design Complexity

Reducing Embedded Software Costs

Reducing Physical Validation Costs

Optimizing Overall System Performance

State-of-the-Art Engineering Practices



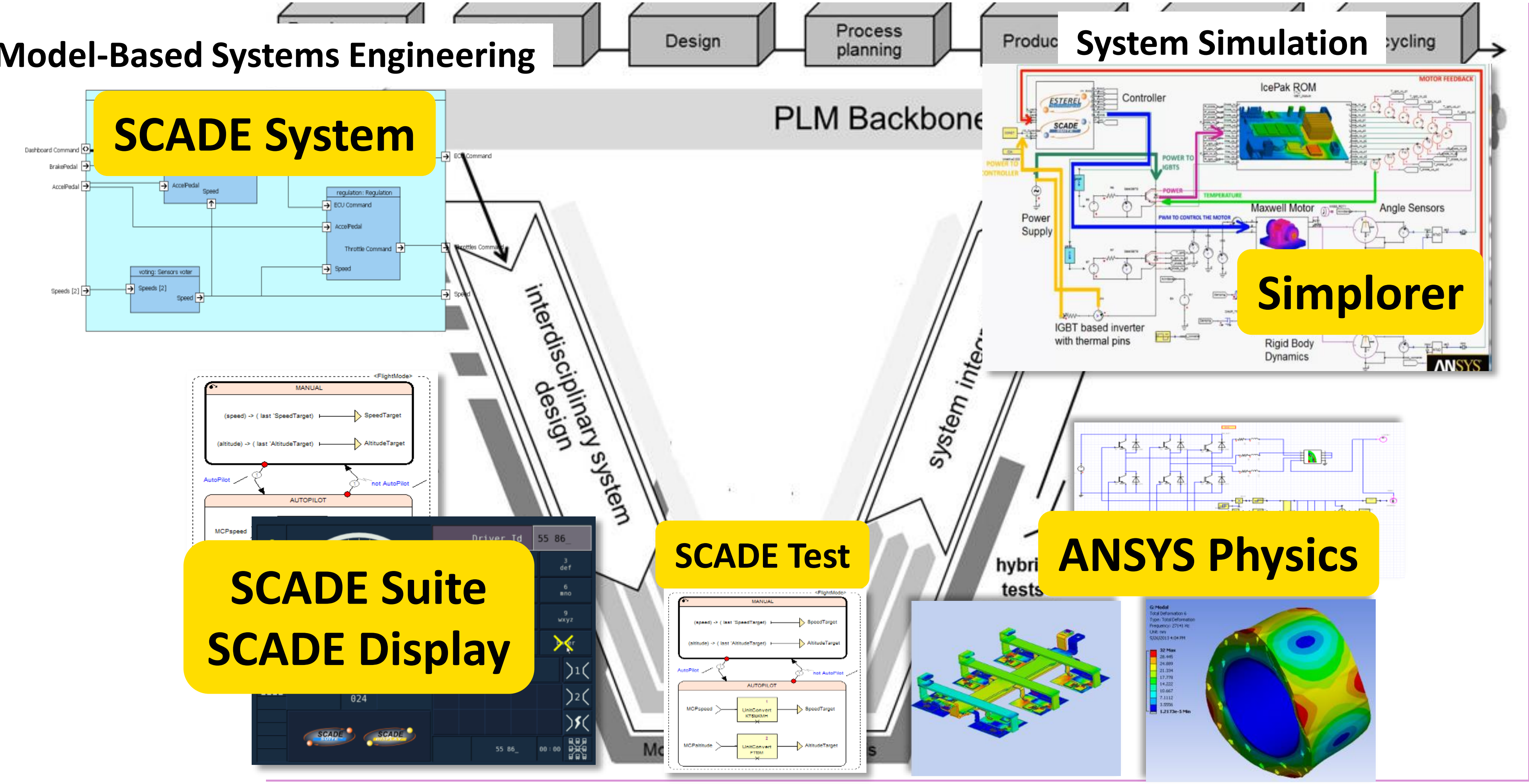
Managing Design Complexity → Model-Based Systems Engineering

Reducing Embedded Software Costs → Model-Based Software Engineering

Reducing Physical Validation Costs → 3D Physics Simulation

Optimizing Overall System Performance → Multi-Physics and System Simulation

ANSYS Model-Based Engineering Solutions



Model-Based Systems Engineering

SCADE System

System Simulation

Simplorer

SCADE Suite
SCADE Display

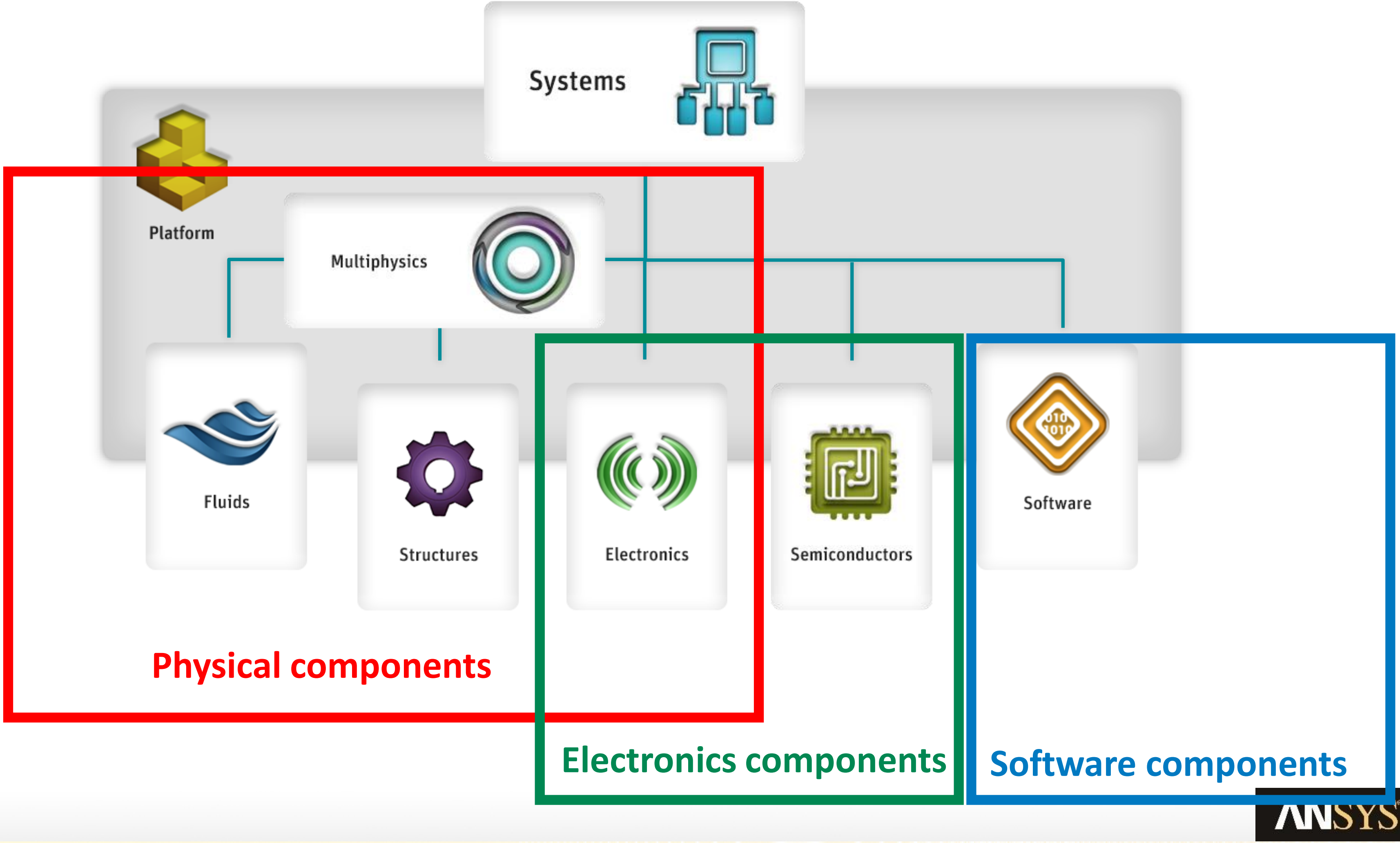
SCADE Test

ANSYS Physics

Model-Based Software Engineering

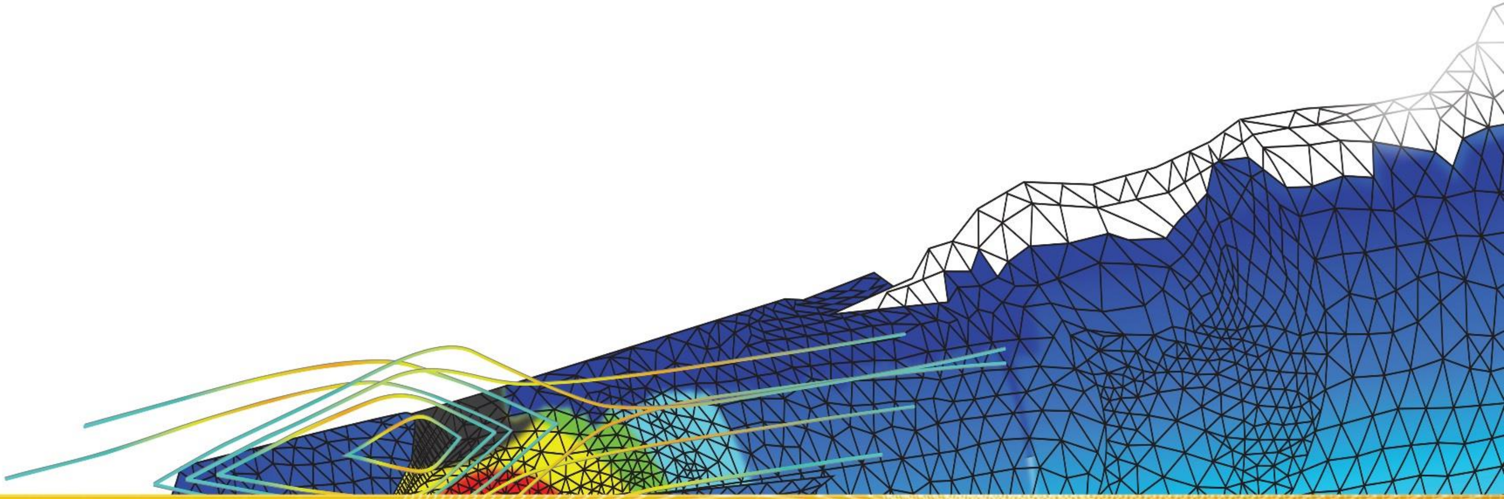
3D Physical Simulation & Multiphysics

ANSYS System Simulation Platform leverages simulation at the component level

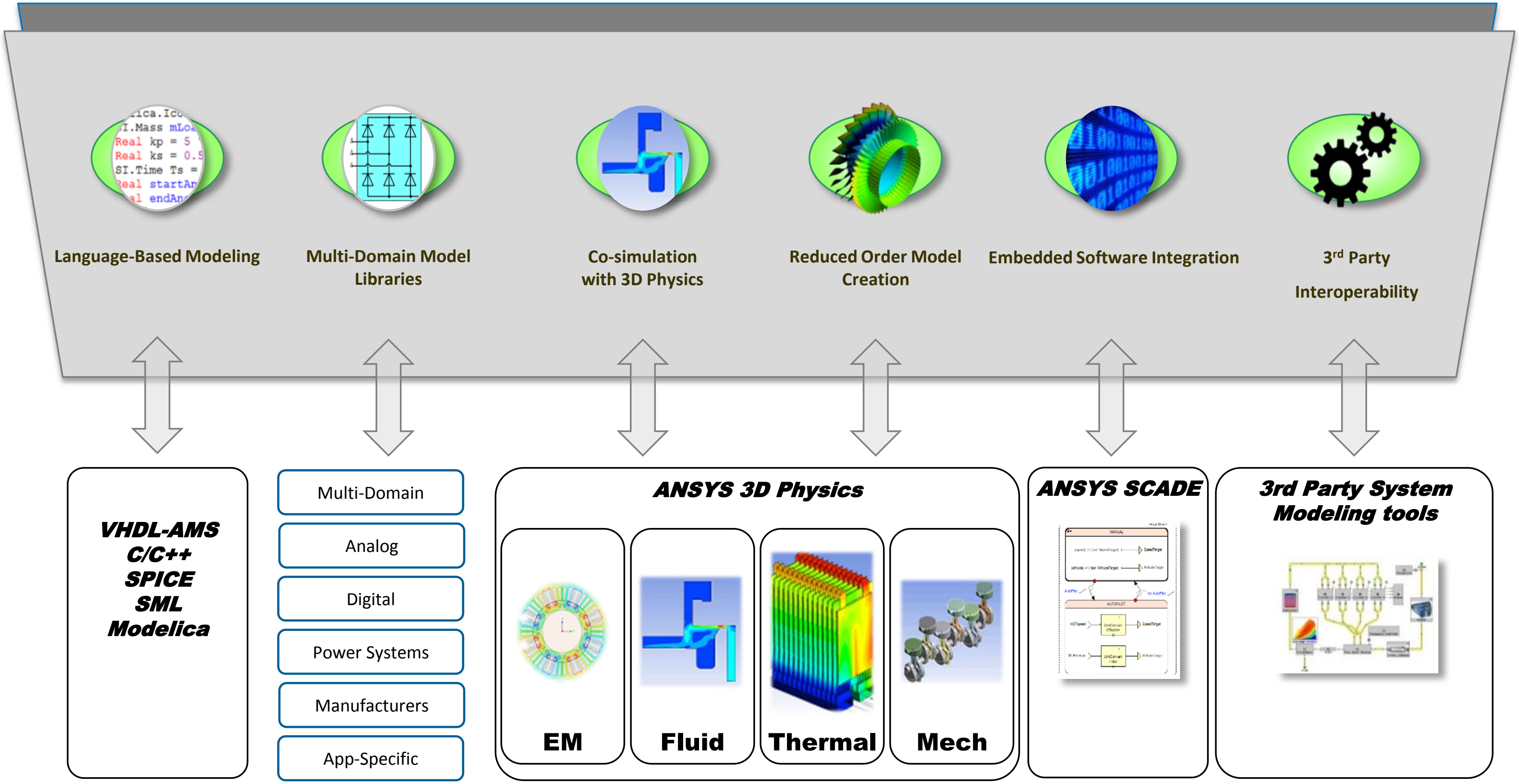


ANSYS®

Simplorer



Modeling the System in Simplorer



Language-Based Modeling

Multi-Domain Model Libraries

Co-simulation with 3D Physics

Reduced Order Model Creation

Embedded Software Integration

3rd Party

Interoperability

VHDL-AMS
C/C++
SPICE
SML
Modelica

- Multi-Domain
- Analog
- Digital
- Power Systems
- Manufacturers
- App-Specific

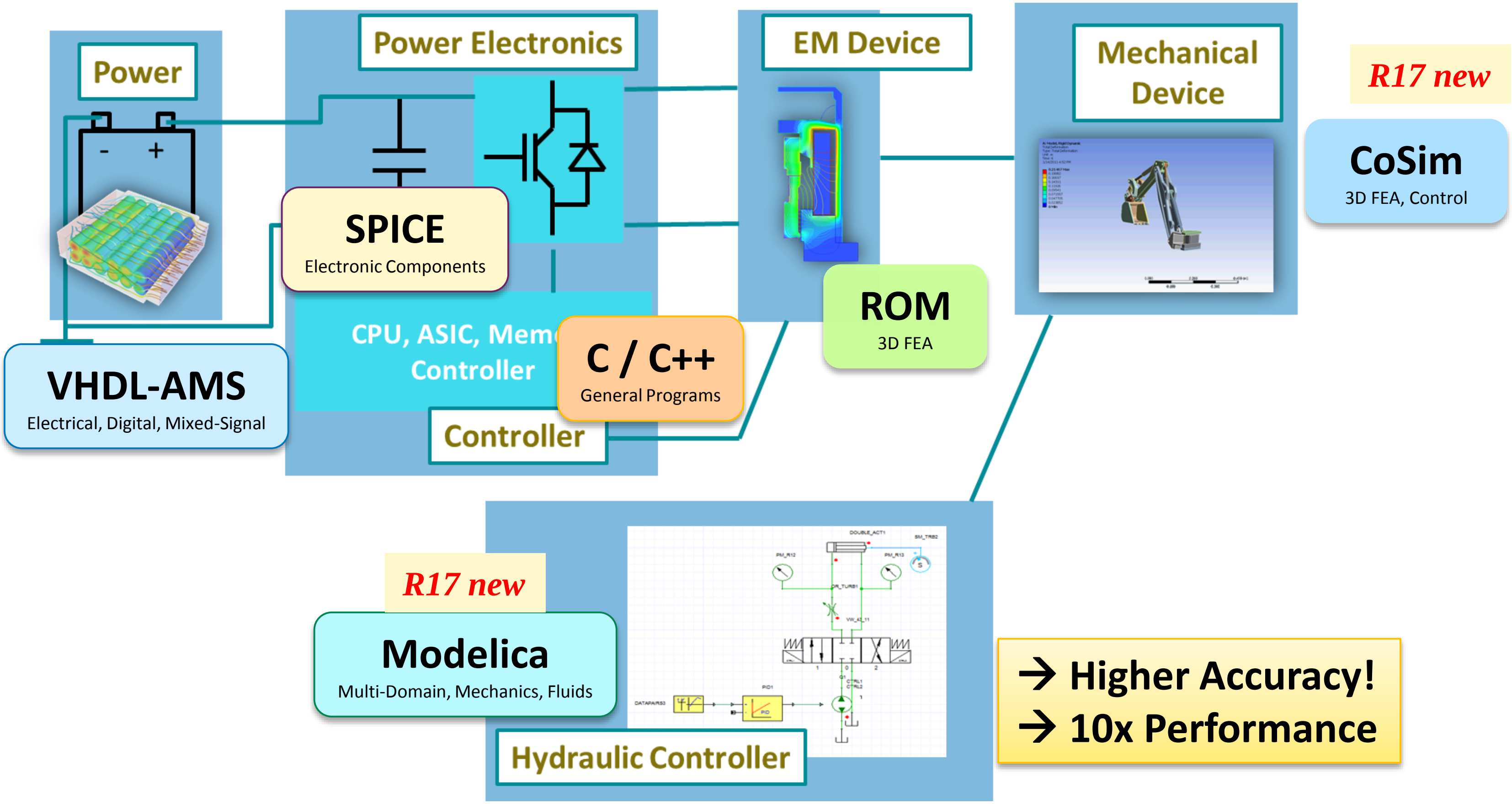
ANSYS 3D Physics

EM | **Fluid** | **Thermal** | **Mech**

ANSYS SCADE

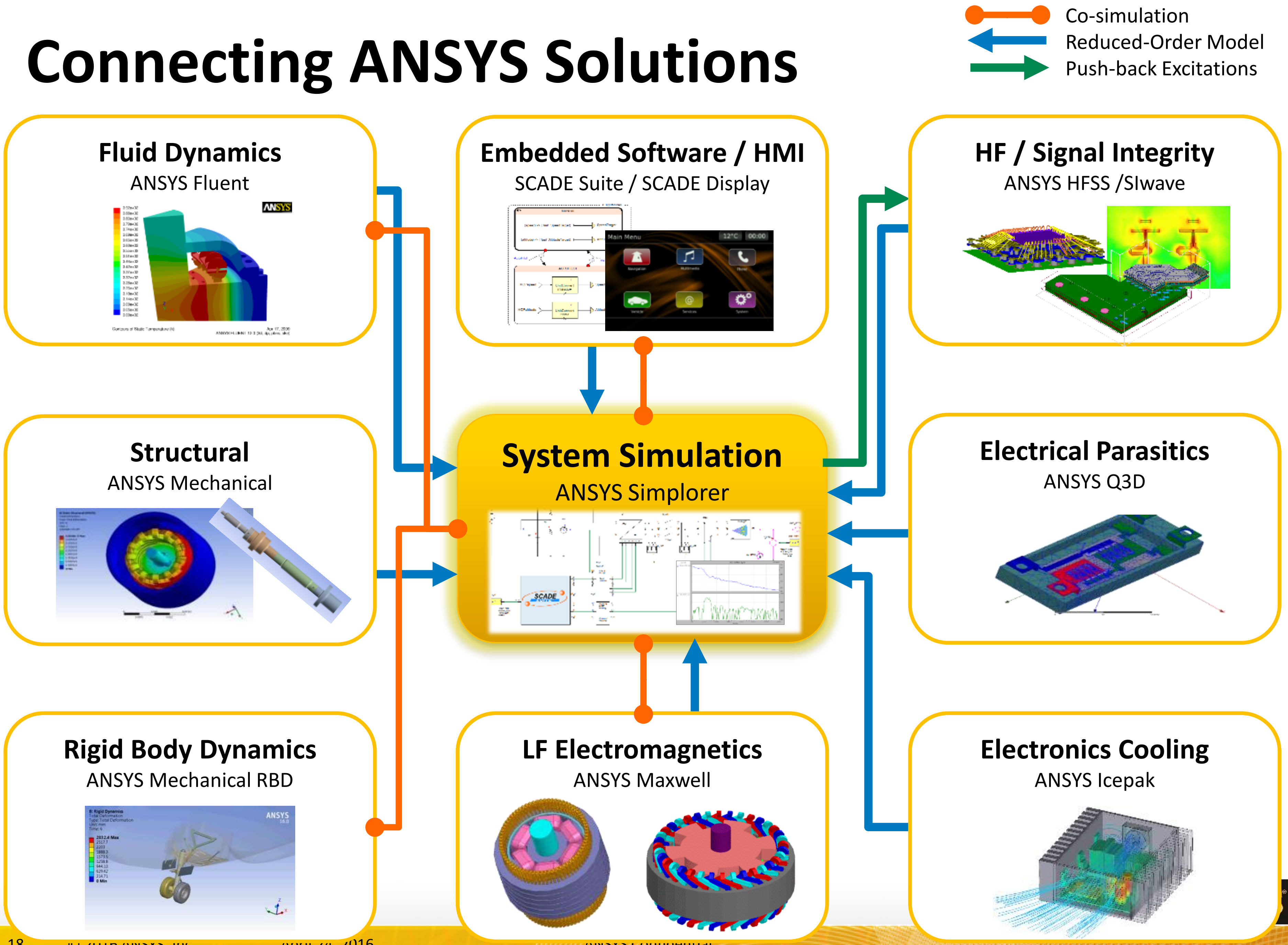
3rd Party System Modeling tools

Modelica: 10x Productivity for Systems Verification



→ Higher Accuracy!
→ 10x Performance

Connecting ANSYS Solutions



Co-simulation

Reduced-Order Model

Push-back Excitations

Fluid Dynamics
ANSYS Fluent

Embedded Software / HMI
SCADE Suite / SCADE Display

HF / Signal Integrity
ANSYS HFSS /SIwave

Structural
ANSYS Mechanical

System Simulation
ANSYS Simplorer

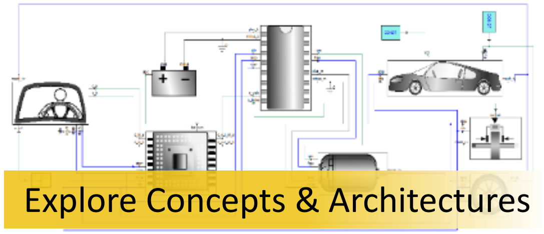
Electrical Parasitics
ANSYS Q3D

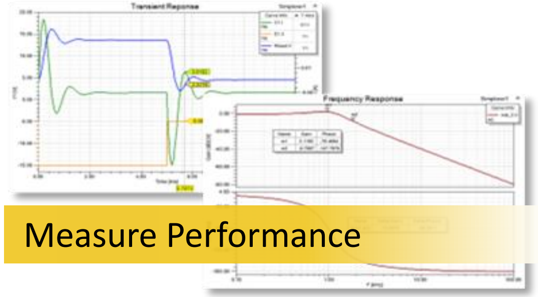
Rigid Body Dynamics
ANSYS Mechanical RBD

LF Electromagnetics
ANSYS Maxwell

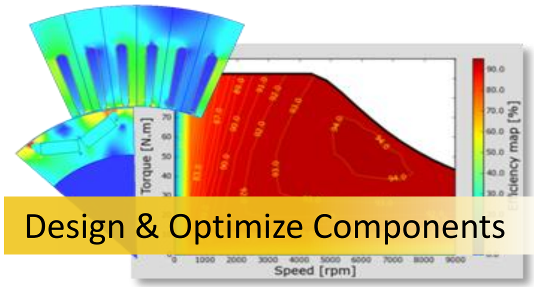
Electronics Cooling
ANSYS Icepak

Multi-Domain, Multi-Fidelity Simulation

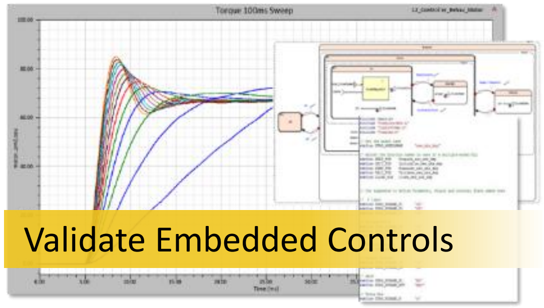

Explore Concepts & Architectures



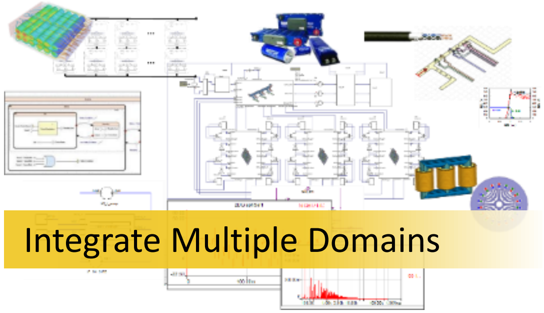
Measure Performance

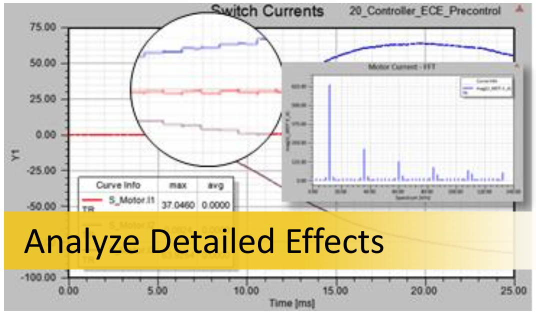


Design & Optimize Components

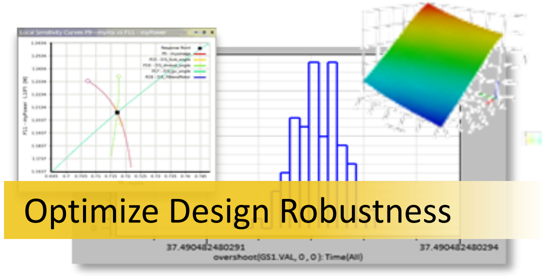


Validate Embedded Controls

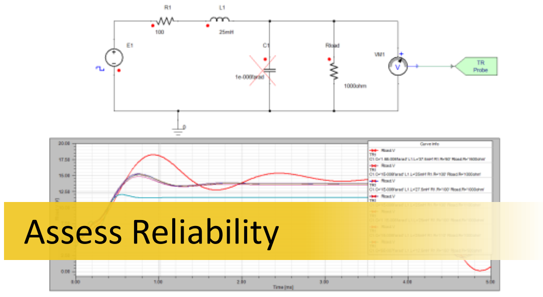

Integrate Multiple Domains



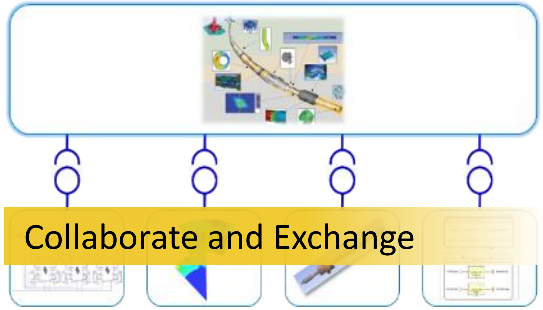
Analyze Detailed Effects



Optimize Design Robustness

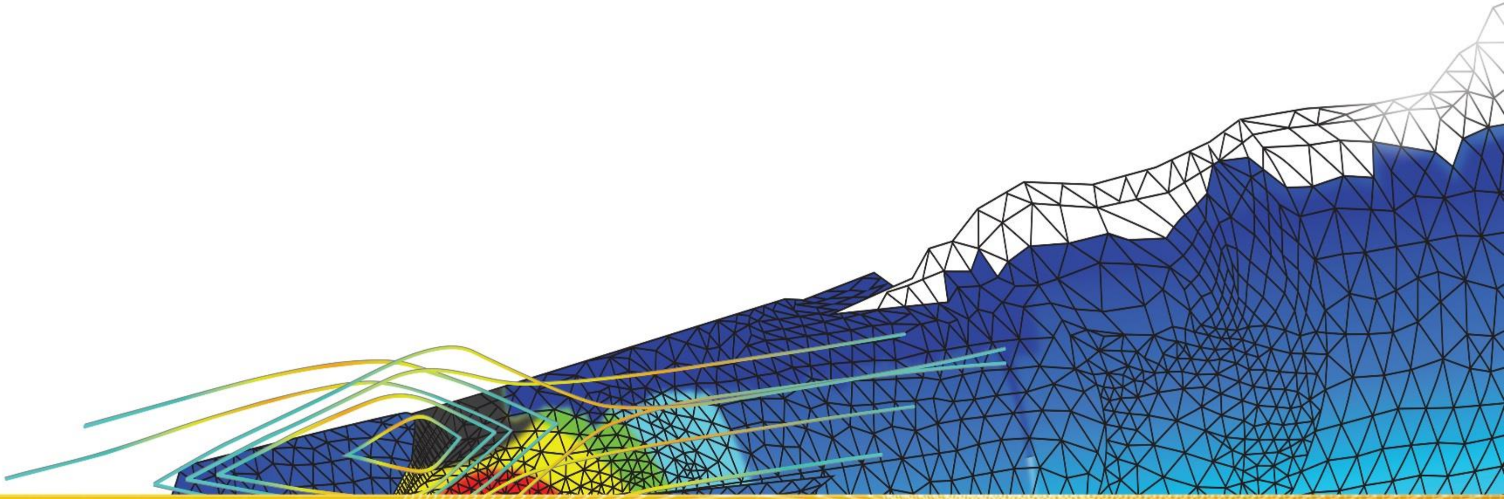


Assess Reliability


Collaborate and Exchange

ANSYS®

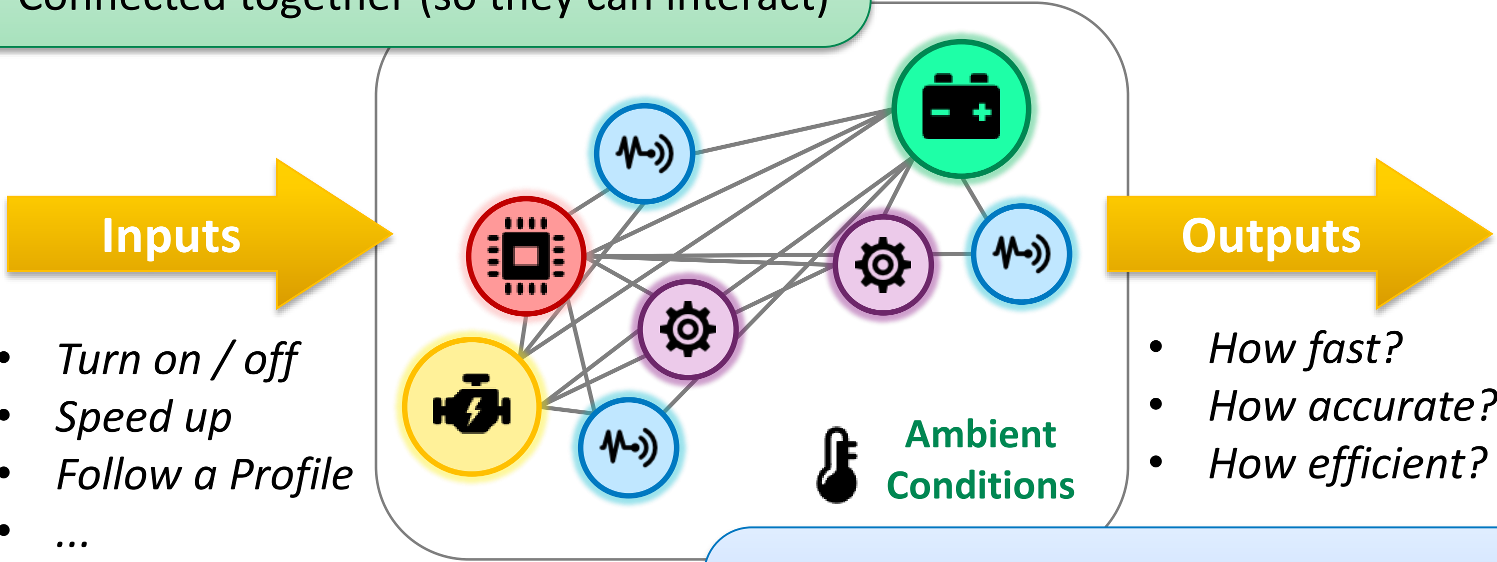
Modeling System Behavior



What Does it Mean to Model & Simulate a System?

System Modeling

- Mathematical descriptions of behavior
- Captured in a formal modeling language
- Connected together (so they can interact)



- *Turn on / off*
- *Speed up*
- *Follow a Profile*
- …

- *How fast?*
- *How accurate?*
- *How efficient?*

System Simulation

- Injects inputs and sets conditions
- Calculates the response of the system
- Produces outputs to evaluate performance

A Quick Aside: Behavioral Modeling

Mathematical Description of Behavior

$$L_f \frac{di_f}{dt} + R_f * i_f = v_f$$

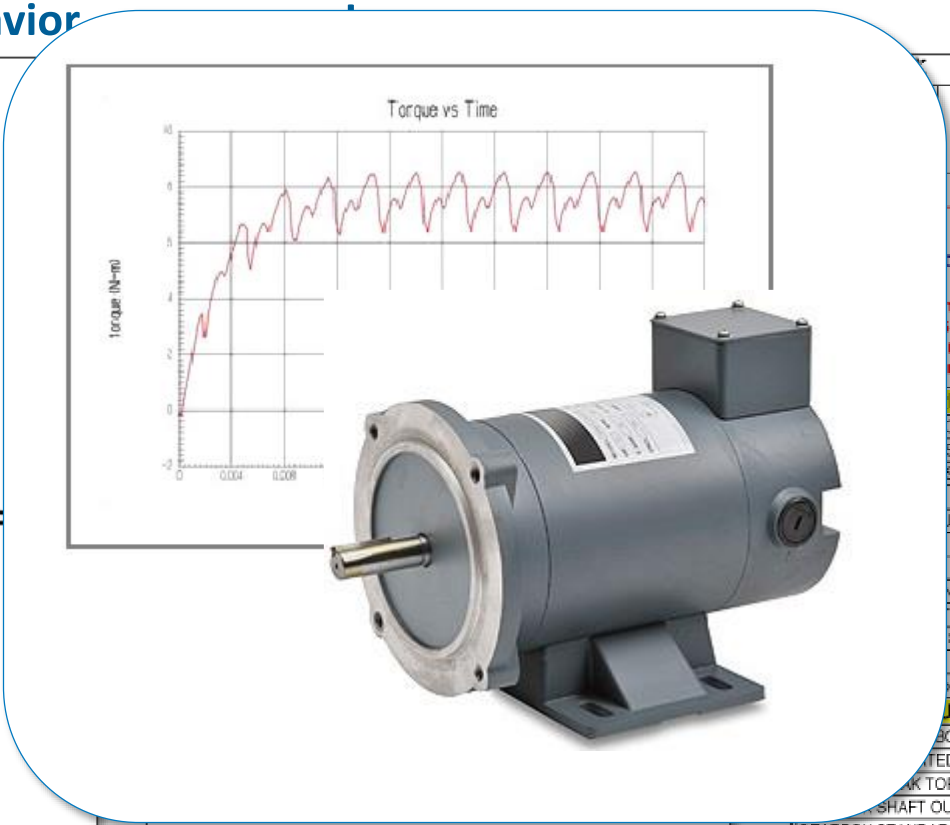
$$I_f = \frac{V_f}{R_f}$$

$$L_a \frac{di_a}{dt} + R_a * i_a + e_a =$$

$$R_a * I_a + E_a = V_a .$$

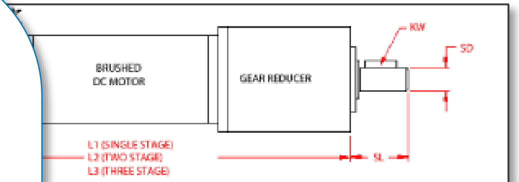
$$E_a = K_m \phi \, w,$$

Expressed in a Formal



```
end architecture behav;
```

Characterized with Empirical Data



NOTE: THE REAR SHAFT EXTENSION PICTURED IS INC
Dimensions; 3/8" Dia by 0.700 Long) Encoders Available

n)
mm)
m
mm)

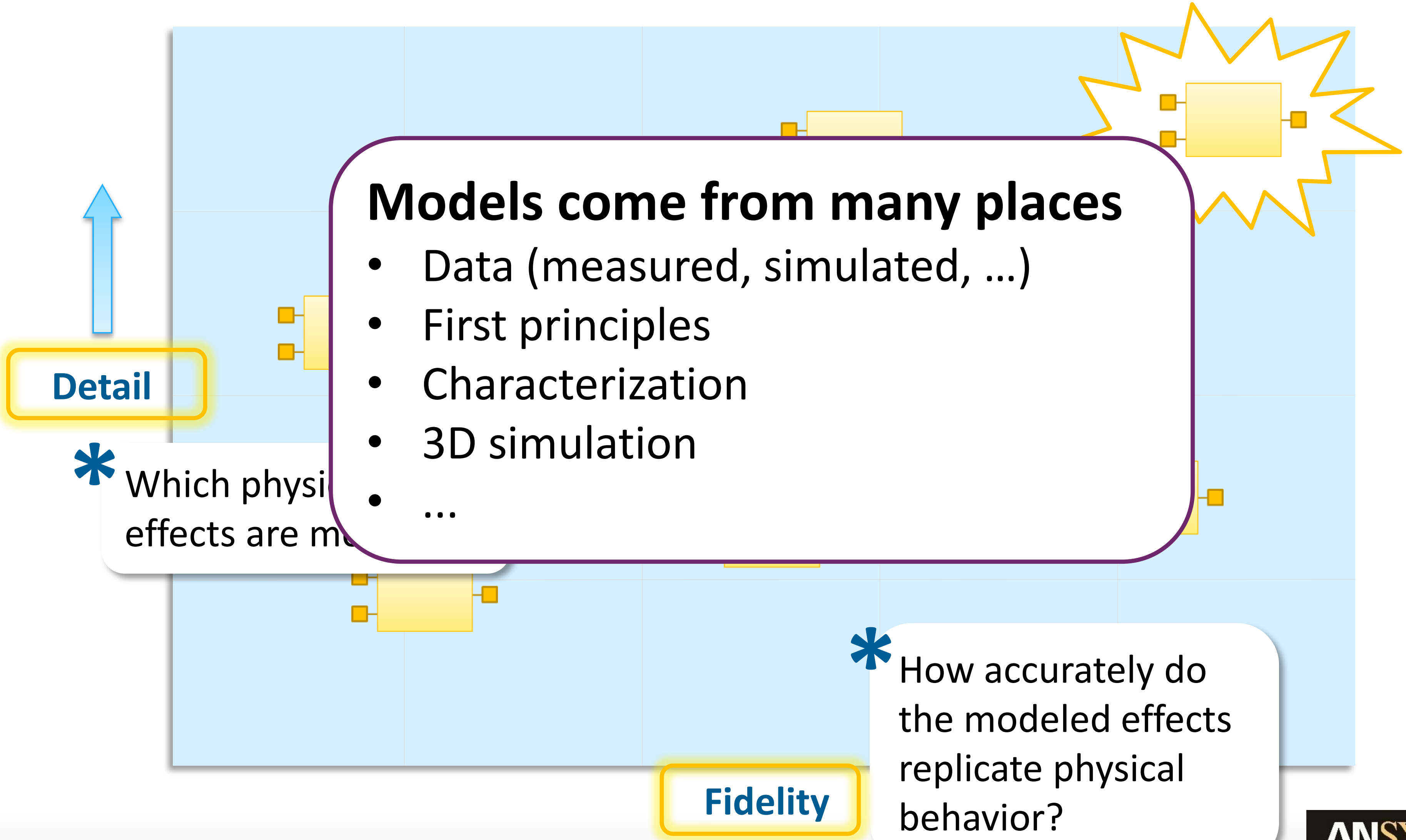
SL= 1.575" (40mm)
GD= 3.2" (81mm)
BD= 1.97" (50mm)
BHC (dia.)=M6x12mm dp (4e

MANCE PARAMETERS	Value
C.	24.0
S CURRENT	13.4
S TORQUE	110
	3500
IS POWER OUT	285
	3800
	0. 79
NT	6.1
	8.2
STANCE AT 1.5 AMPS	0.3
	155
r-only)	1100
UPUT PARAMETERS:	
BOX RATIO	35:1
TED CONTINOUS TORQUE	180
K TORQUE CAPACITY	800
SHAFT OUTPUT SPEED (@ F.L.)	100
GEARBOX STANDARD BACKLASH	30
GEARBOX EFFICIENCY	75%
GEARMOTOR TOTAL WEIGHT	9.0

The Hard Part: Models

- Who/what creates it?
- What detail does it contain? At what fidelity?
- What are the assumptions / limitations?
- What is its interface?
- How accurate is it?
- How is it validated?
- How does it perform?
- How stable is it?
- …

Model Detail & Fidelity



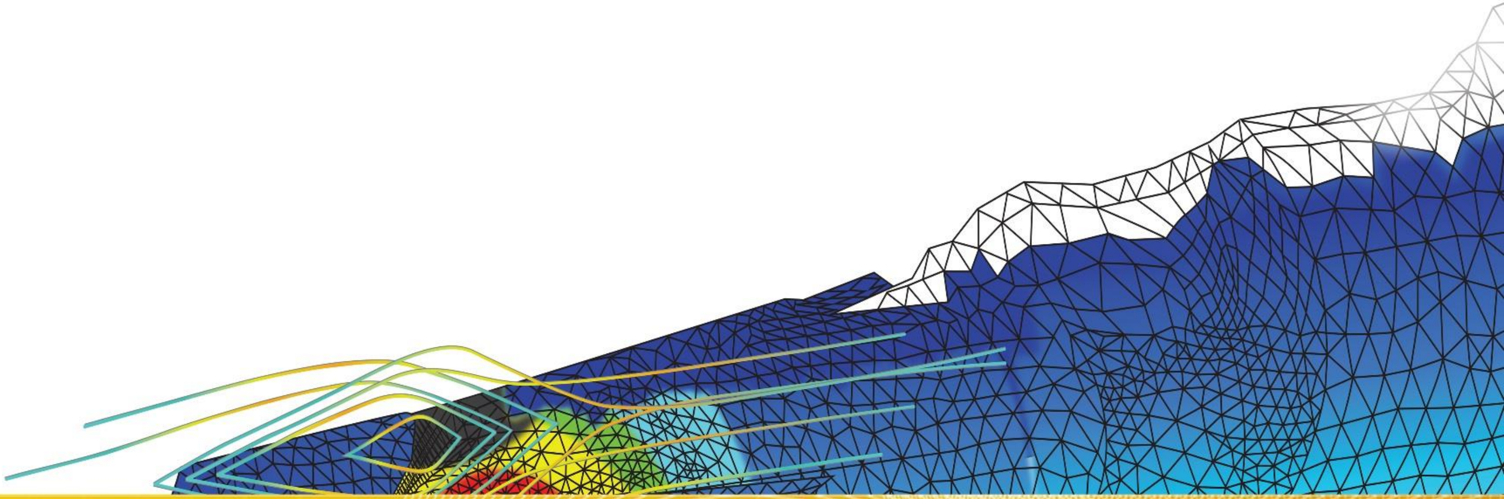
Models come from many places

- Data (measured, simulated, …)
- First principles
- Characterization
- 3D simulation
- …

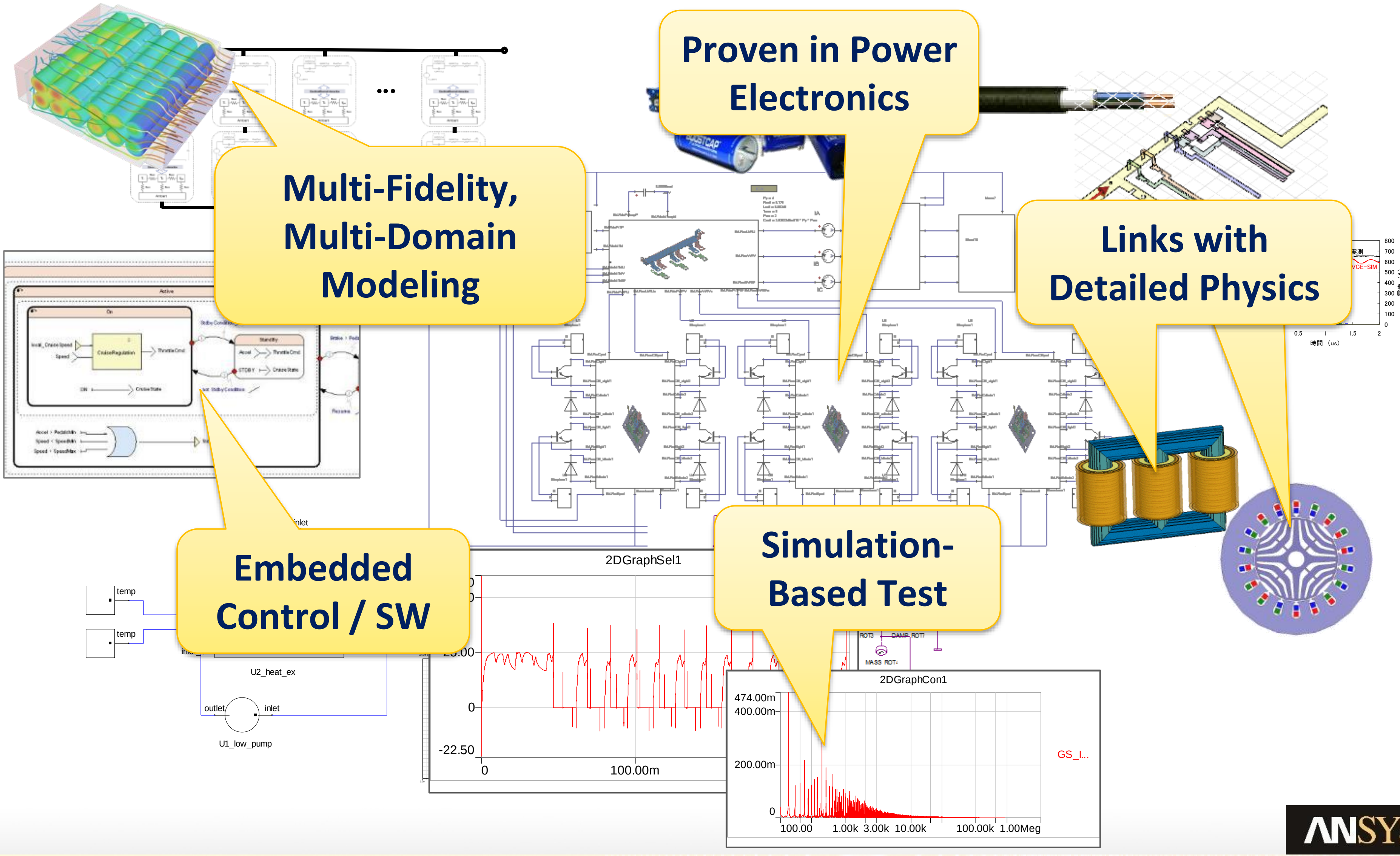
* Which physi effects are m

* How accurately do the modeled effects replicate physical behavior?

Simplorer – Core Capabilities



Simplorer – Physical Modeling & Simulation



Multi-Fidelity, Multi-Domain Modeling

Proven in Power Electronics

Links with Detailed Physics

Embedded Control / SW

Simulation-Based Test

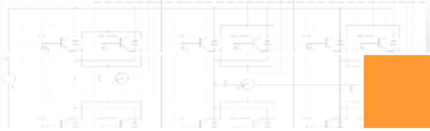
Complementary Solutions: 3D and Systems

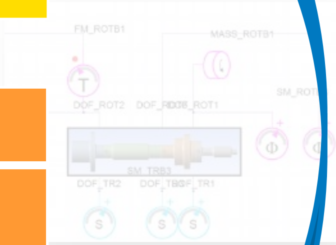
System

Sub-System

Component

	System Simulation
What is simulated?	Integrated assemblies of components – all powered, actuated, controlled and sensed together
Models	Equation-based descriptions of behavior
Size / Complexity	10s to 1,000s of physical, electrical and/or software component models
Simulation	Single solver used for all domains
ANSYS Solution	*Simplorer + Model Libraries + ROMs*



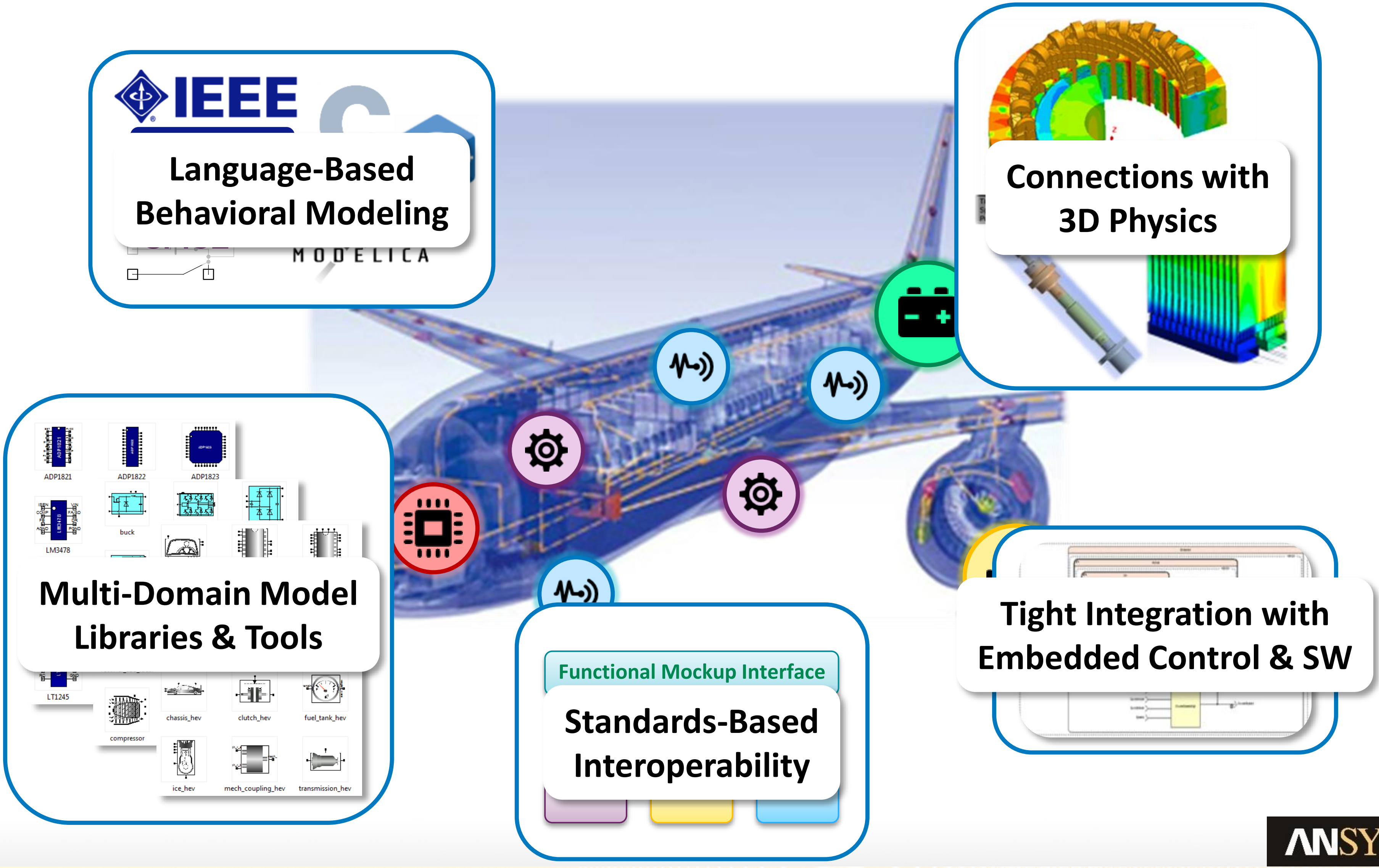


	3D Multiphysics Component Simulation
What is simulated?	Singular components or sub-assemblies – typically focused on the “pure” physics
Models	Geometries, meshed into cells / elements
Size / Complexity	1,000s to 1,000,000s of cells / elements
Simulation	Different solvers for each physics domain, coupled by a simulation platform
ANSYS Solution	*Workbench + 3D Solvers*

mal Software

Modeling Flexibility, Reusability, Interoperability

Essential to Virtual Prototyping



Language-Based Behavioral Modeling

Connections with 3D Physics

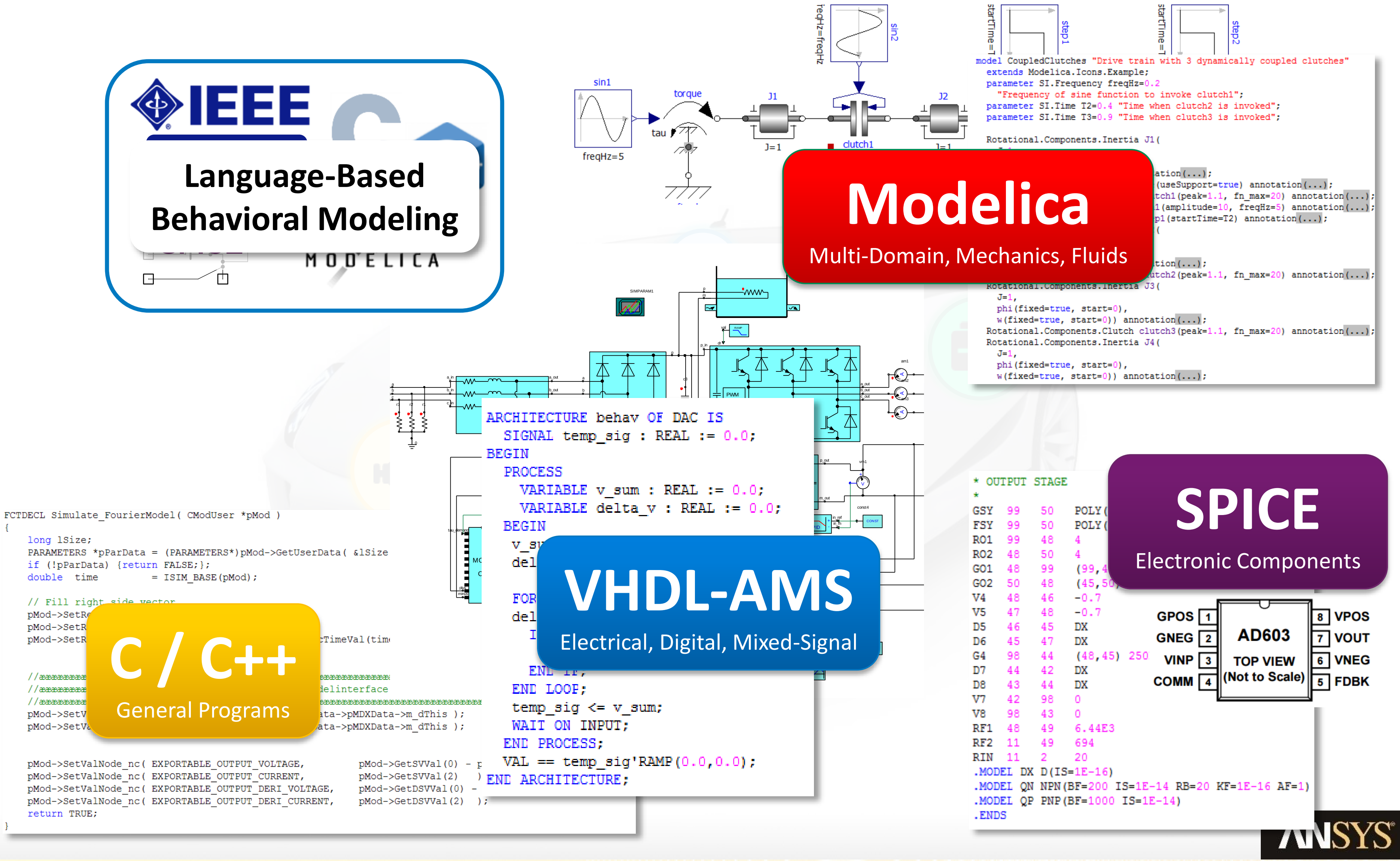
Multi-Domain Model Libraries & Tools

Functional Mockup Interface

Standards-Based Interoperability

Tight Integration with Embedded Control & SW

Languages Common to Design Disciplines



Language-Based Behavioral Modeling

Modelica

Multi-Domain, Mechanics, Fluids

```
model CoupledClutches "Drive train with 3 dynamically coupled clutches"
  extends Modelica.Icons.Example;
  parameter SI.Frequency freqHz=0.2
    "Frequency of sine function to invoke clutch1";
  parameter SI.Time T2=0.4 "Time when clutch2 is invoked";
  parameter SI.Time T3=0.9 "Time when clutch3 is invoked";

  Rotational.Components.Inertia J1(
                                     ation(...);
                                     (useSupport=true) annotation(...);
                                     tch1(peak=1.1, fn_max=20) annotation(...);
                                     1(amplitude=10, freqHz=5) annotation(...);
                                     p1(startTime=T2) annotation(...);
                                     (

                                     tion(...);
                                     utch2(peak=1.1, fn_max=20) annotation(...);
  Rotational.Components.Inertia J3(
    J=1,
    phi(fixed=true, start=0),
    w(fixed=true, start=0)) annotation(...);
  Rotational.Components.Clutch clutch3(peak=1.1, fn_max=20) annotation(...);
  Rotational.Components.Inertia J4(
    J=1,
    phi(fixed=true, start=0),
    w(fixed=true, start=0)) annotation(...);
```

VHDL-AMS

Electrical, Digital, Mixed-Signal

```
ARCHITECTURE behav OF DAC IS
  SIGNAL temp_sig : REAL := 0.0;
BEGIN
  PROCESS
   VARIABLE v_sum : REAL := 0.0;
   VARIABLE delta_v : REAL := 0.0;
  BEGIN
   v_su
   del

   FOR
   del
     I

     END IF;
   END LOOP;
   temp_sig <= v_sum;
   WAIT ON INPUT;
  END PROCESS;
 VAL == temp_sig'RAMP(0.0,0.0);
END ARCHITECTURE;
```

SPICE

Electronic Components

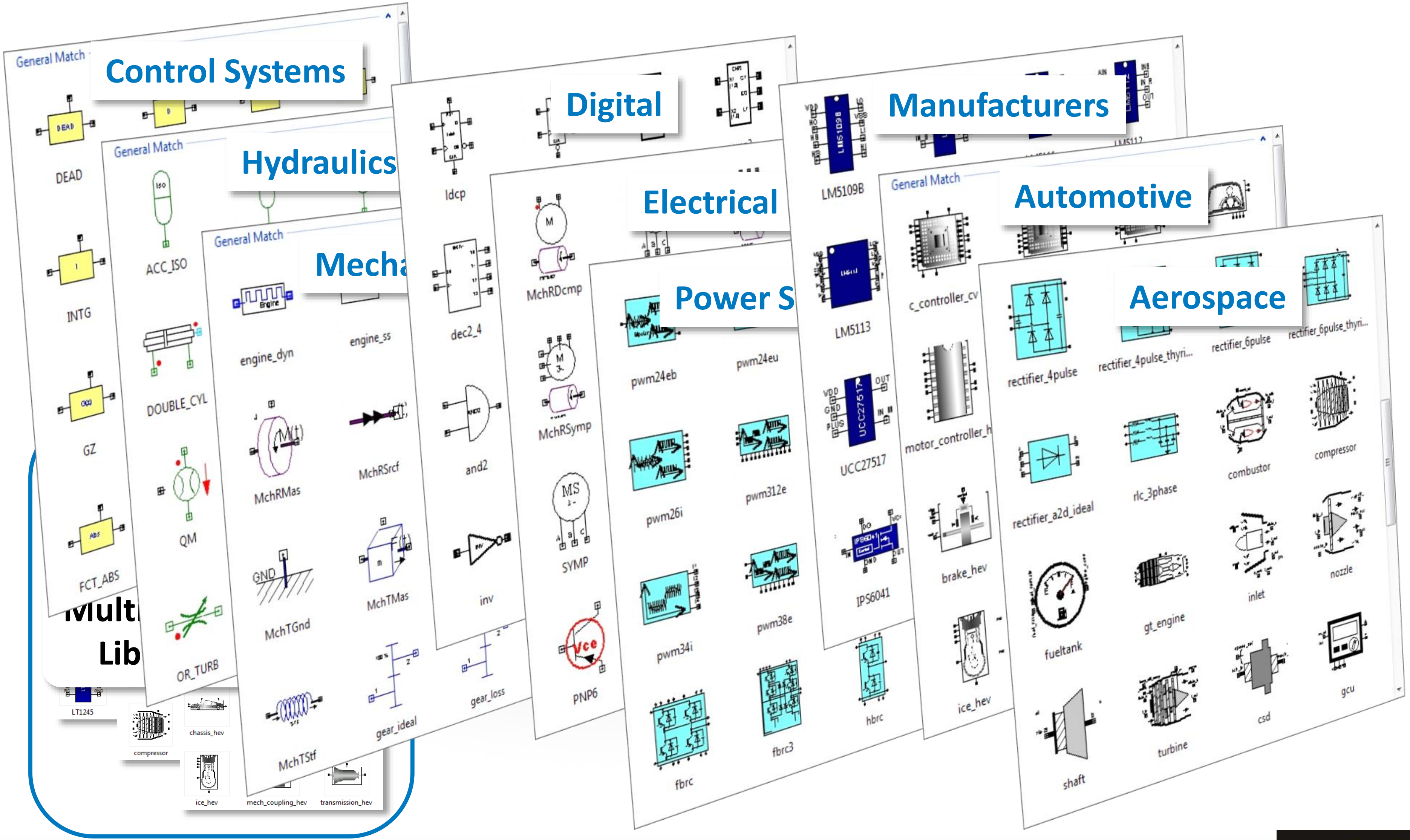
```
* OUTPUT STAGE
*
GSY  99   50   POLY(
FSY  99   50   POLY(
RO1  99   48   4
RO2  48   50   4
GO1  48   99   (99,4
GO2  50   48   (45,50
V4   48   46   -0.7
V5   47   48   -0.7
D5   46   45   DX
D6   45   47   DX
G4   98   44   (48,45) 250
D7   44   42   DX
D8   43   44   DX
V7   42   98   0
V8   98   43   0
RF1  48   49   6.44E3
RF2  11   49   694
RIN  11   2    20
.MODEL DX D(IS=1E-16)
.MODEL QN NPN(BF=200 IS=1E-14 RB=20 KF=1E-16 AF=1)
.MODEL QP PNP(BF=1000 IS=1E-14)
.ENDS
```

C / C++

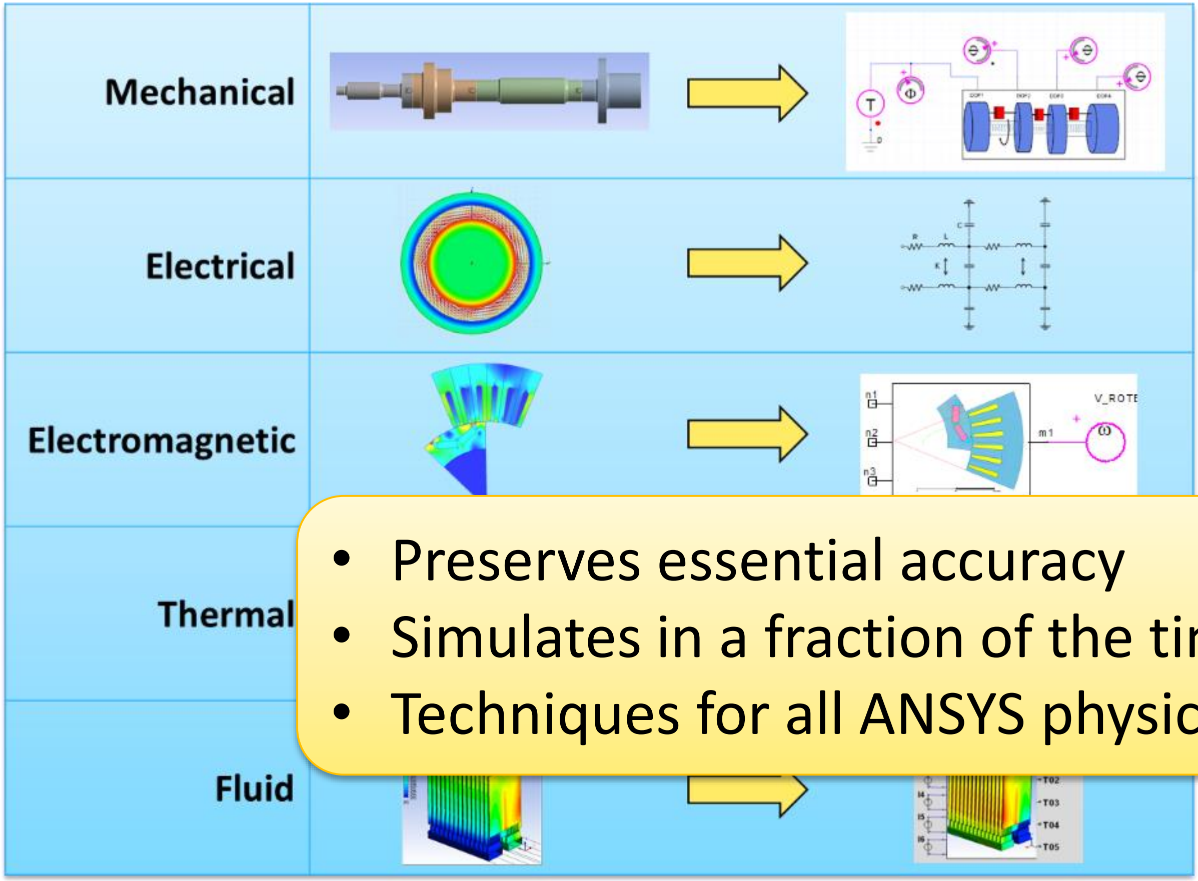
General Programs

```
FCTDECL Simulate_FourierModel( CModUser *pMod )
{
    long lSize;
    PARAMETERS *pParData = (PARAMETERS*)pMod->GetUserData( &lSize
    if (!pParData) {return FALSE;};
    double  time          = ISIM_BASE(pMod);

    // Fill right side vector
    pMod->SetRe
    pMod->SetR
    pMod->SetR                                              cTimeVal(time


    //æææææææææ                                             æææææææææææææææ
    //æææææææææ                                             delinterface
    //æææææææææ                                             æææææææææææææææææææææææææææææææ
    pMod->SetV                                              ata->pMDXData->m_dThis );
    pMod->SetVa                                             ata->pMDXData->m_dThis );


    pMod->SetValNode_nc( EXPORTABLE_OUTPUT_VOLTAGE,         pMod->GetSVVal(0) - p
    pMod->SetValNode_nc( EXPORTABLE_OUTPUT_CURRENT,         pMod->GetSVVal(2)   )
    pMod->SetValNode_nc( EXPORTABLE_OUTPUT_DERI_VOLTAGE,    pMod->GetDSVVal(0) -
    pMod->SetValNode_nc( EXPORTABLE_OUTPUT_DERI_CURRENT,    pMod->GetDSVVal(2)   );
    return TRUE;
}
```

Model Libraries for Multi-Domain Systems



Reduced-Order Modeling (ROM) Interfaces



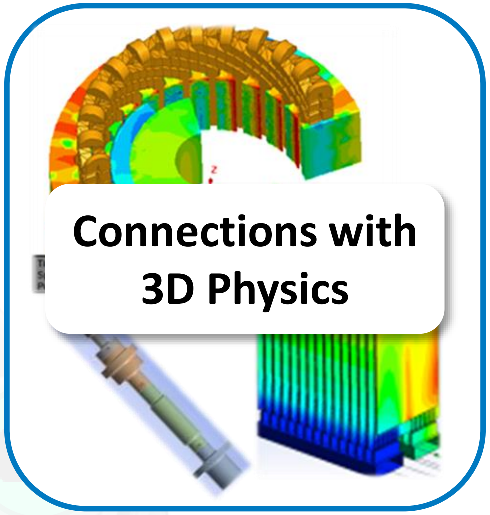
Mechanical

Electrical

Electromagnetic

Thermal

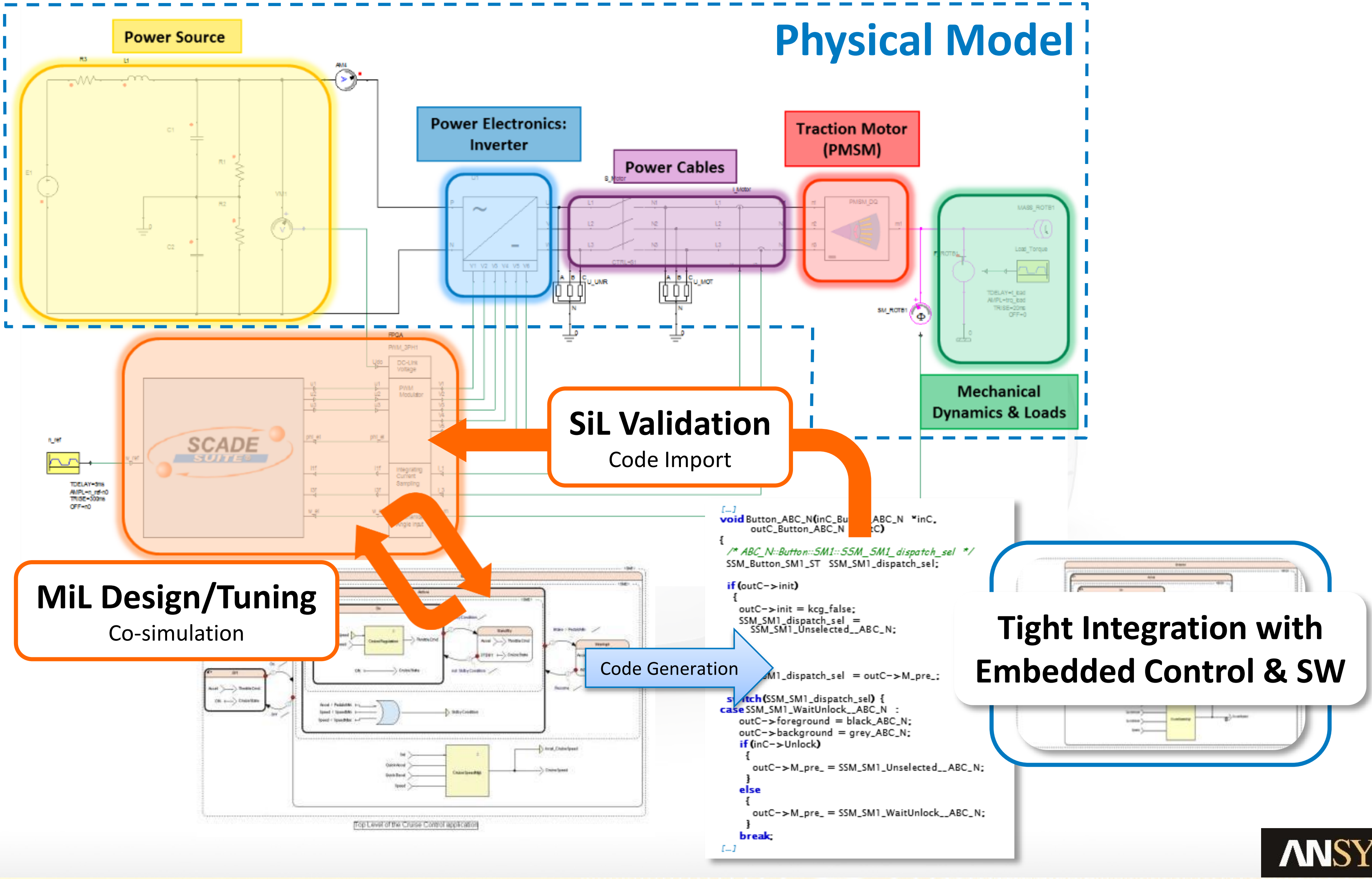
Fluid



Connections with 3D Physics

- Preserves essential accuracy
- Simulates in a fraction of the time required by 3D
- Techniques for all ANSYS physics

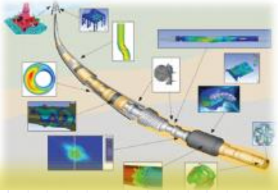
Physical Modeling for Embedded Control


Physical Model
Power Source
Power Electronics: Inverter
Power Cables
Traction Motor (PMSM)
Mechanical Dynamics & Loads
SCADE SUITE
SiL Validation
Code Import
MiL Design/Tuning
Co-simulation
Code Generation
Tight Integration with Embedded Control & SW
Top Level of the Cruise Control application

```
[…]
void Button_ABC_N(inC_Bu      ABC_N  *inC,
     outC_Button_ABC_N       C)
{
  /* ABC_N::Button::SM1::SSM_SM1_dispatch_sel  */
  SSM_Button_SM1_ST  SSM_SM1_dispatch_sel;

  if (outC->init)
   {
    outC->init = kcg_false;
    SSM_SM1_dispatch_sel  =
      SSM_SM1_Unselected__ABC_N;



     M1_dispatch_sel  = outC->M_pre_;

  s   itch (SSM_SM1_dispatch_sel) {
case SSM_SM1_WaitUnlock__ABC_N  :
    outC->foreground = black_ABC_N;
    outC->background = grey_ABC_N;
    if (inC->Unlock)
     {
      outC->M_pre_ = SSM_SM1_Unselected__ABC_N;
     }
    else
     {
      outC->M_pre_ = SSM_SM1_WaitUnlock__ABC_N;
     }
    break;
[…]
```

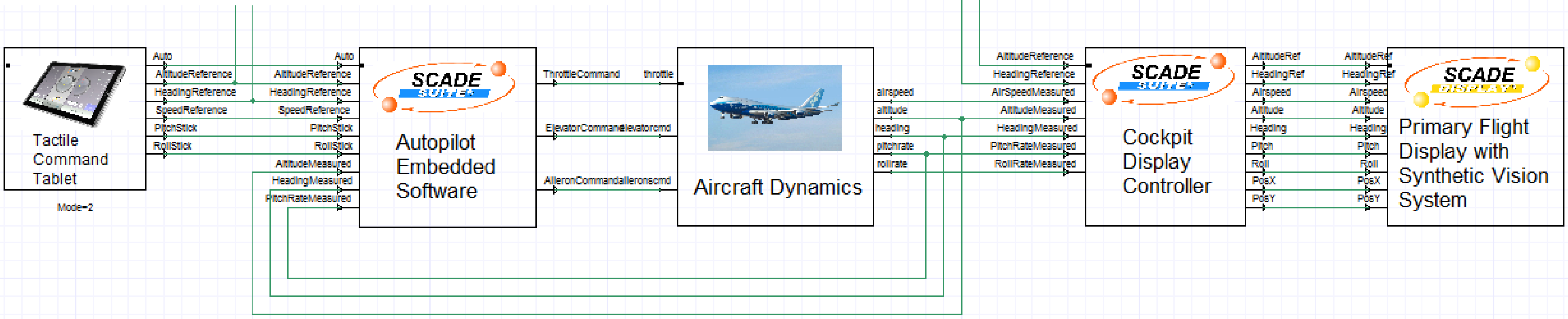
Standards-Based Interoperability

FMI for Integrated System Simulation



MODEL PORTABILITY
TOOL INTEROPERABILITY
ENTERPRISE DEPLOYABILITY

Example: Flight Control System



Functional Mockup Interface

Standards-Based Interoperability