Model-Based Design and Code Generation

Tom Erkkinen, Embedded Applications
tom.erkkinen@mathworks.com +1 248-596-7943

July 2004

Agenda

- Challenges
- Solutions
- Examples
- Summary

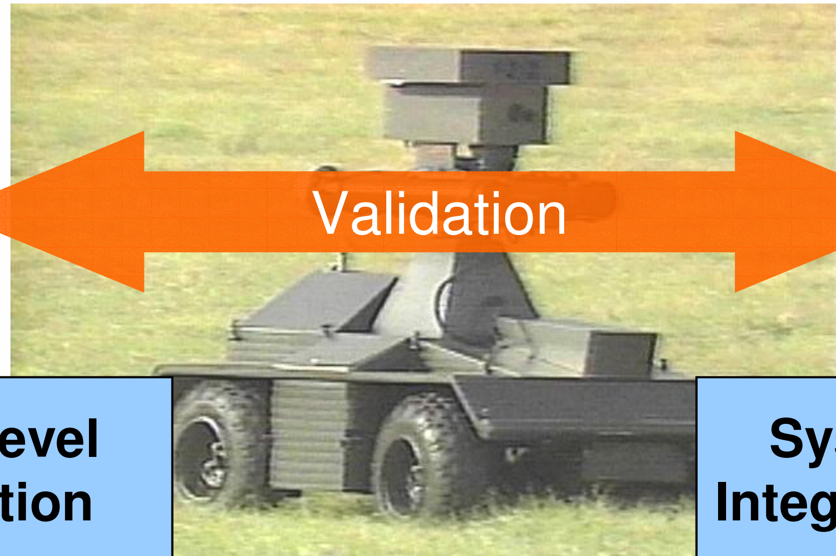
Challenges

- **Shorter development time**
  - Political and business climate
  - Increasing design complexity
  - Changing threats and tactics
  - Wartime needs
- **Team integration**
  - Digital hardware, DSP S/W, Control S/W teams, Flight Dynamics
  - All use different tools and communicate via written documents
  - Contractor/subcontractor/ supplier/customer relationships
- **Paradigm shift towards “COTS”**
- **Rising importance of electronics and software**

Technical Design Challenges

- **Increasingly large, complex, interdependent systems**
  - More functionality, interoperability requirements
  - Requirement for partitioning tasks, communicating ideas, integrating results
- **Interfacing components**
  - How to evaluate impact of component or algorithm change on system?
- **Evaluating component performance during design**
  - How to explore design tradeoffs before implementation?
- **HIL/Rapid prototyping**
  - Pinpoint errors before they propagate into hardware
  - Get working prototypes to the warfighter quickly
  - Rapidly iterate design(s)
  - Understand real-time performance implications before hardware is available

Defense System Development



Validation

Define Requirements

System-Level Specification

Subsystem Design

Subsystem Implementation

Subsystem Integration & Test

System-Level Integration & Test

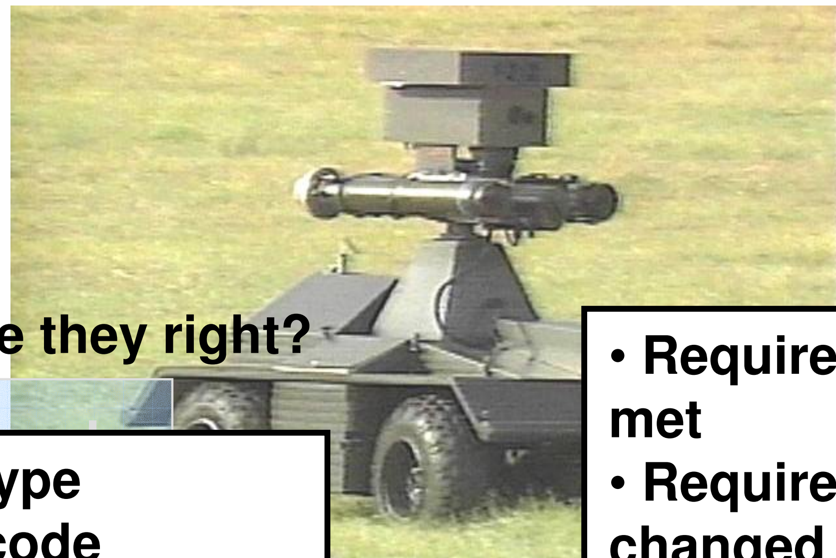
Integration, Test and Calibration

Traditional process is document-based and cumbersome

Similar processes:

- Spiral Development
- Waterfall
- Etc.

Typical Development Process Failure Points



Define Requirements

REQ

Are they right?

- Build prototype product and code
- Create written design specification

SPEC + Errors

Manually Write Code

CODE + Errors

~~Integration, Test and Calibration~~

- Requirements not met
- Requirements have changed
- System doesn't work as desired

Principal Problems and Difficulties Associated with Military Software Development*

Problem #1: Software is inherently complex

Problem #2: Poor estimates of cost, schedule, and size

Problem #3: Unstable requirements

Problem #4: Poor problem-solving/decision-making

Problem #5: Belief in the silver bullets to cure our ills

Problem #6: Failure to obtain in-depth awareness of the program

*http://www.nps.navy.mil/wings/acq_topics/PrincipalProblems.html

May 18-19, 2004

Software Tools Forum

Federal Aviation Administration

EMBRY-RIDDLE AERONAUTICAL UNIVERSITY

Commercial aviation faces similar challenges

Untitled Document - Microsoft Internet Explorer provided by The Mathworks, Inc.

1:45-2:10	Verification Tool Research Sponsored by FAA	Hayhurst (NASA) Santhanam, Waldrop, Chilenski (Boeing)
2:10-2:55	Use of the MathWorks Tool Suite to Develop DO178B Certified Code	Potter (Honeywell)
2:55-3:25	*Break*	
3:25-4:05	Tool Qualification - A Living Process	Roth (Honeywell)
4:05-4:30	Structural Coverage Analysis for Level A	Romanski (Verocel)
4:30-4:55	Model-Based Analysis and Test Generation for Complex Systems	Busser, Blackburn, Nauman (T-VEC Technologies)
4:55-5:00	Wrap up	
5:00-	*Reception*	

Wednesday, May 19, 2004

Time	Topic	Speaker
8:00-8:15	Overview of the Day	Rierson (FAA)
8:15-8:45	How Not To Get 'Bitten' By Tool Qualification	Palmer, Fine (Honeywell)
8:45-9:00	Verification Tool Qualification: A Commercial Tool Vendor's Perspective of the FAA Qualification Process	Reeve (Patmos) Whipple (Metroweks)
9:00-9:15	TTP-Verify	Schwarz (TTTech)
9:15-9:30	Experience Using an Automated Analysis/Code Coverage Tool for a Joint Strike Fighter Component and Space Shuttle Flight Software Verification Support Software	Badgley, Davis (GBTech)
9:30-10:00	*Break*	
	Process and Emerging Technology	
10:00-10:40	Tool Intensive Software Development: New Challenges for Verification, Validation, & Certification	Heimdahl (Univ. of Minnesota)
10:40-11:20	Embedded System Architecture Analysis Using the SAE AADL	Feiler, Hudak (SEI) Gluch (ERAU)
11:20-11:35	Using AADL for Safety and Security Features	Colbert (Univ. of South California) Land (High Integrity Solutions)
11:35-11:50	Reusing Tool Qualification Data: CAST Perspective (Draft Paper)	Rierson (FAA)
11:50-12:10	Application of the Reusable Software Component (RSC) Guidance to the Qualification of a Software Verification Tool	Waldrop, Martz, Santhanam (Boeing)
12:10-1:30	*Lunch*	
1:30-1:45	The Simulink/Stateflow Analyzer (SSA)	Galloway, Toyn, Iwu, McDermid (U. of York)
1:45-2:00	A Revolution in Avionics Software Safety Verification Traceability	O'Leary (Verocel)
2:00-2:15	Using Tool Service History for Tool Qualification	Petesch (Hamilton Sunstrand)
2:15-2:30	A High-Productivity Tool for Developing Complex Safey-Critical Software	Crocker (Escher Technologies)
2:30-3:00	Software Verification with Emerging Technologies	Thornton, Frey (Honeywell)

Solutions

No Slide Title - Microsoft Internet Explorer provided by The Mathworks, Inc.

File Edit View Favorites Tools Help

Back Links Address D:\Home\tech mktg\events presentations\faaforum\faa\Potter_files\frame Go

Benefits of using COTS tools for model based development

- Model based development puts the emphasis on improving quality of the design
  - Error rates for design errors are still relatively high compared to implementation errors, model base development emphasizes design error prevention and detection
  - Error rates for code and compilation are relatively low, zero defects is achievable when safe subsets are used for COTS code generators and compilers
- Achieves two important industry goals
  - Reduces cost
  - Improves safety

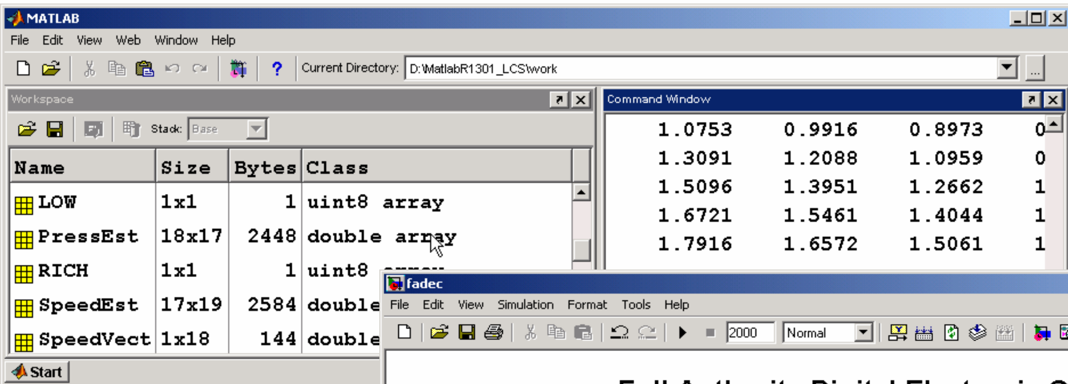
May 2004 Bill Potter Honeywell 7

Outline Slide 7 of 15 Slide Show

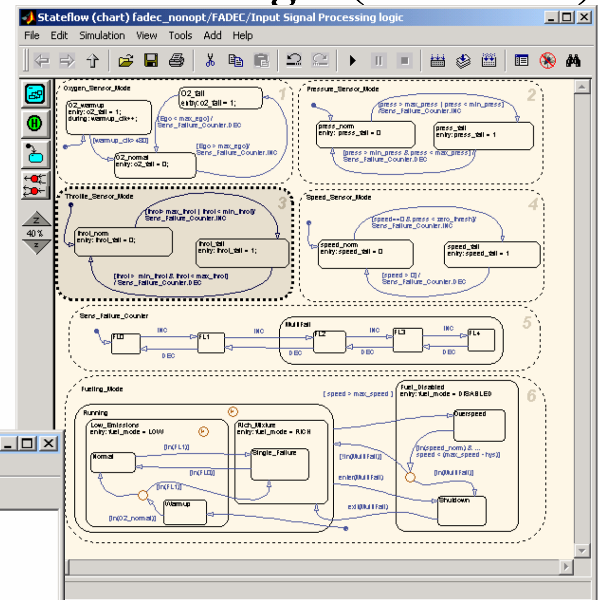
Done My Computer

A Model

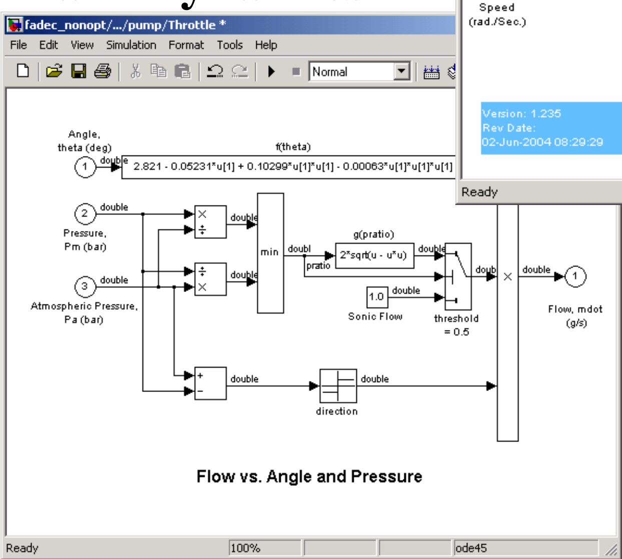
Data

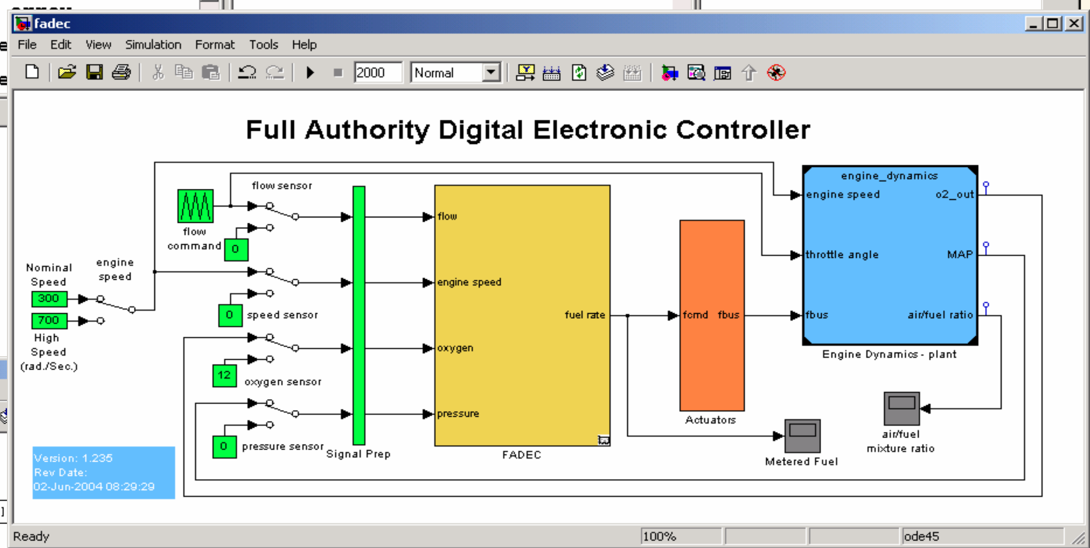


Control Logic (Stateflow)

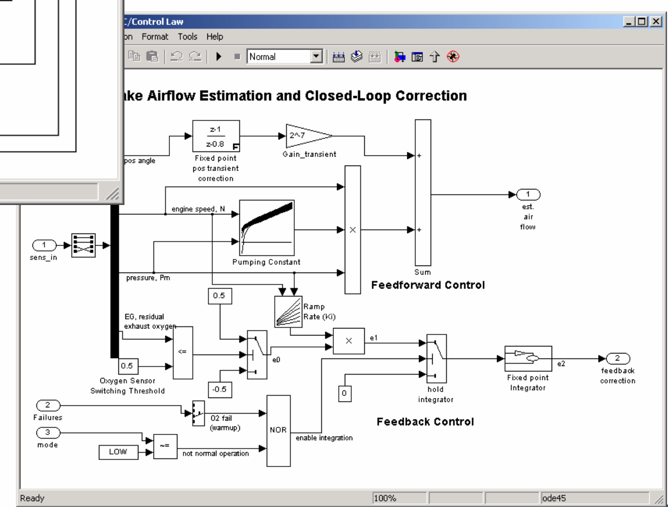


Plant Dynamics



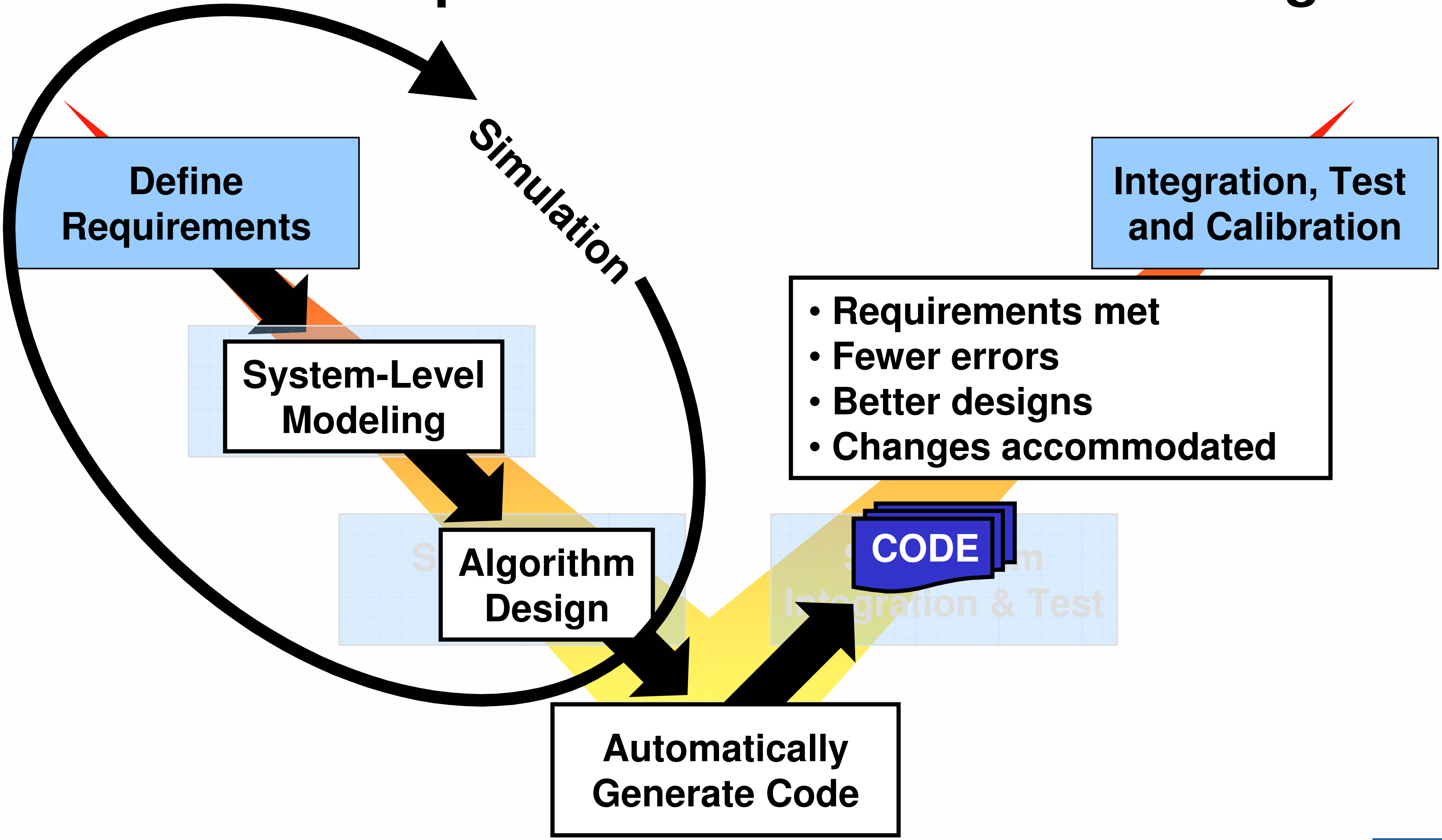


Feedback Control

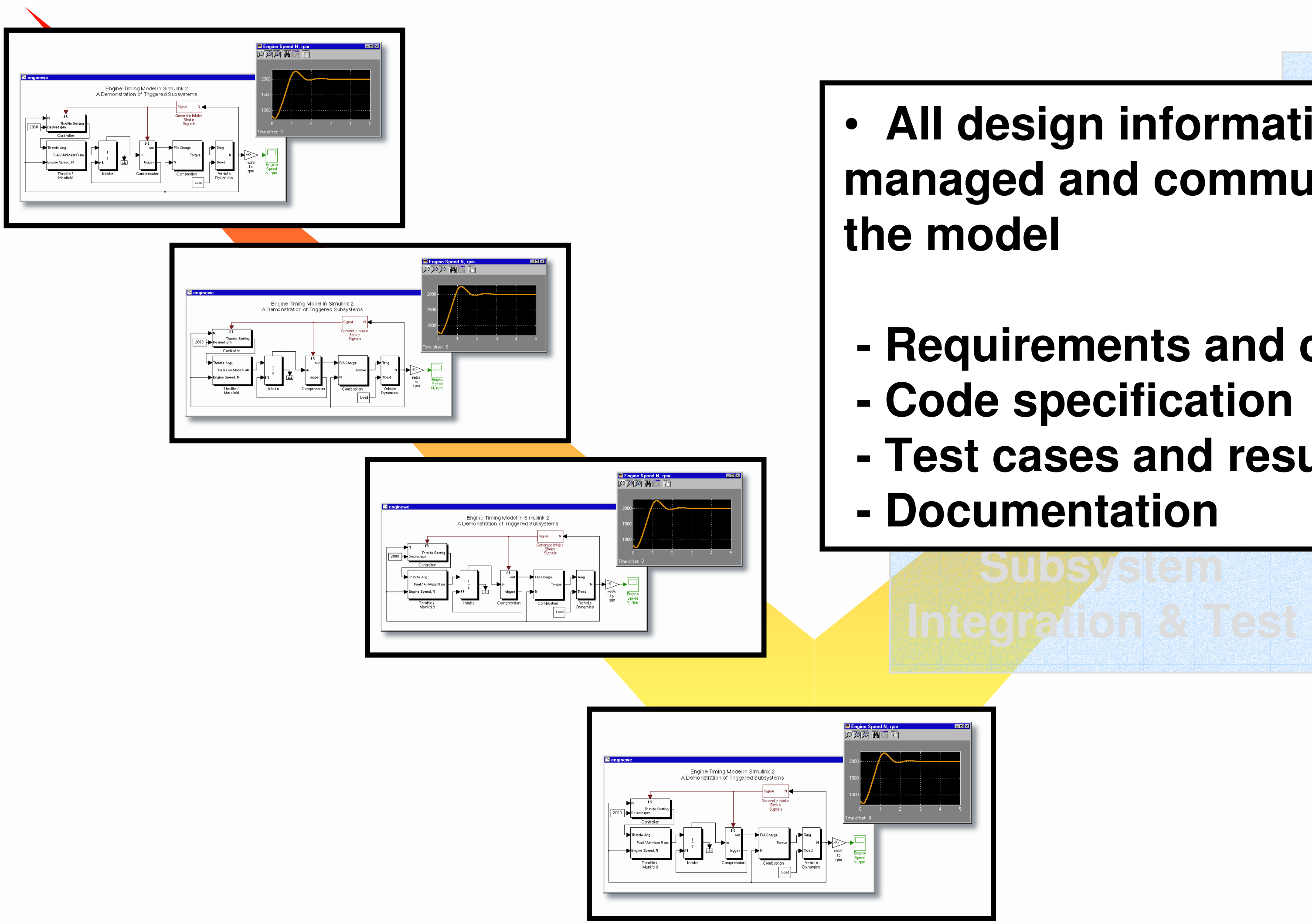


System Model

Process is Improved with Model-Based Design



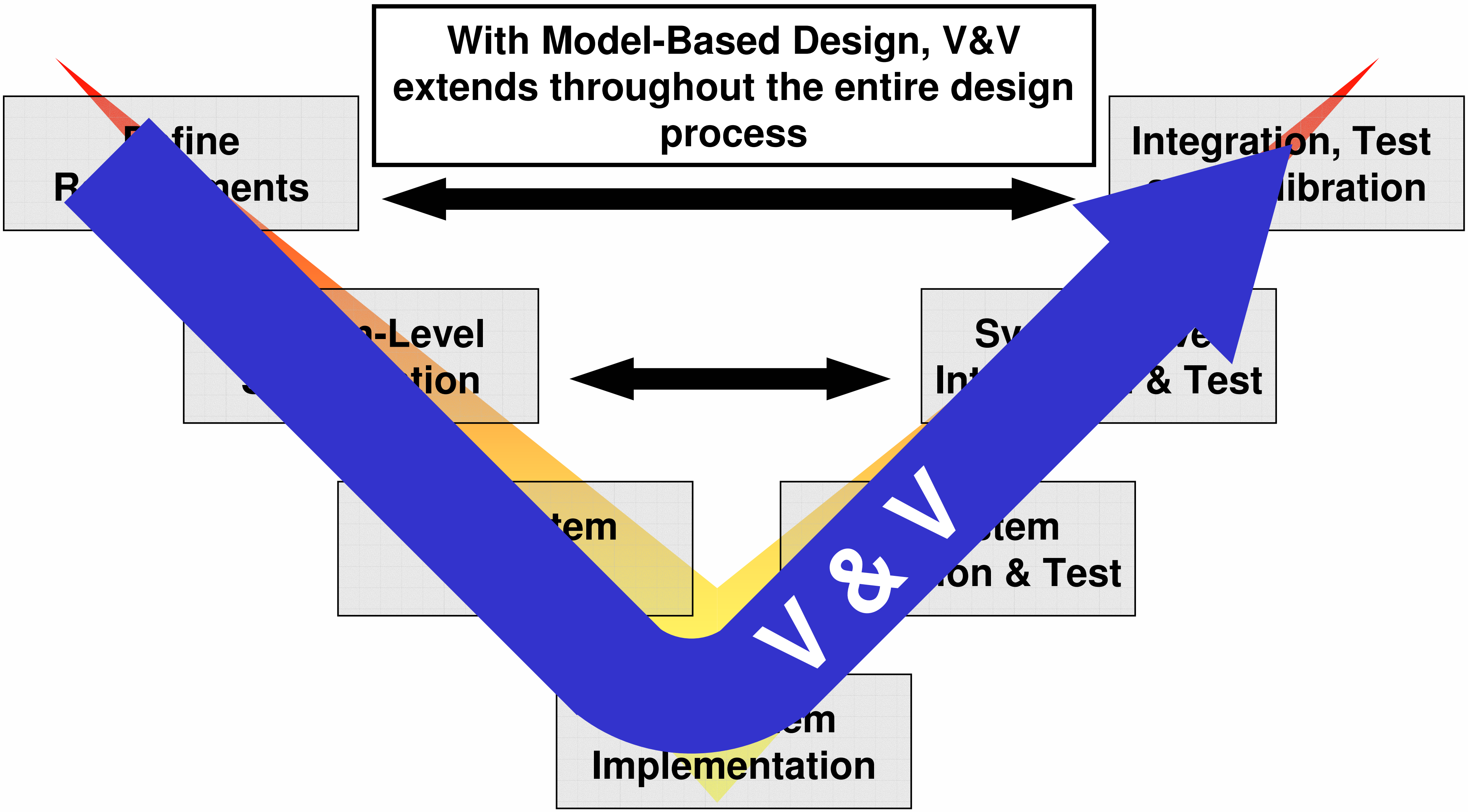
Model Information Management



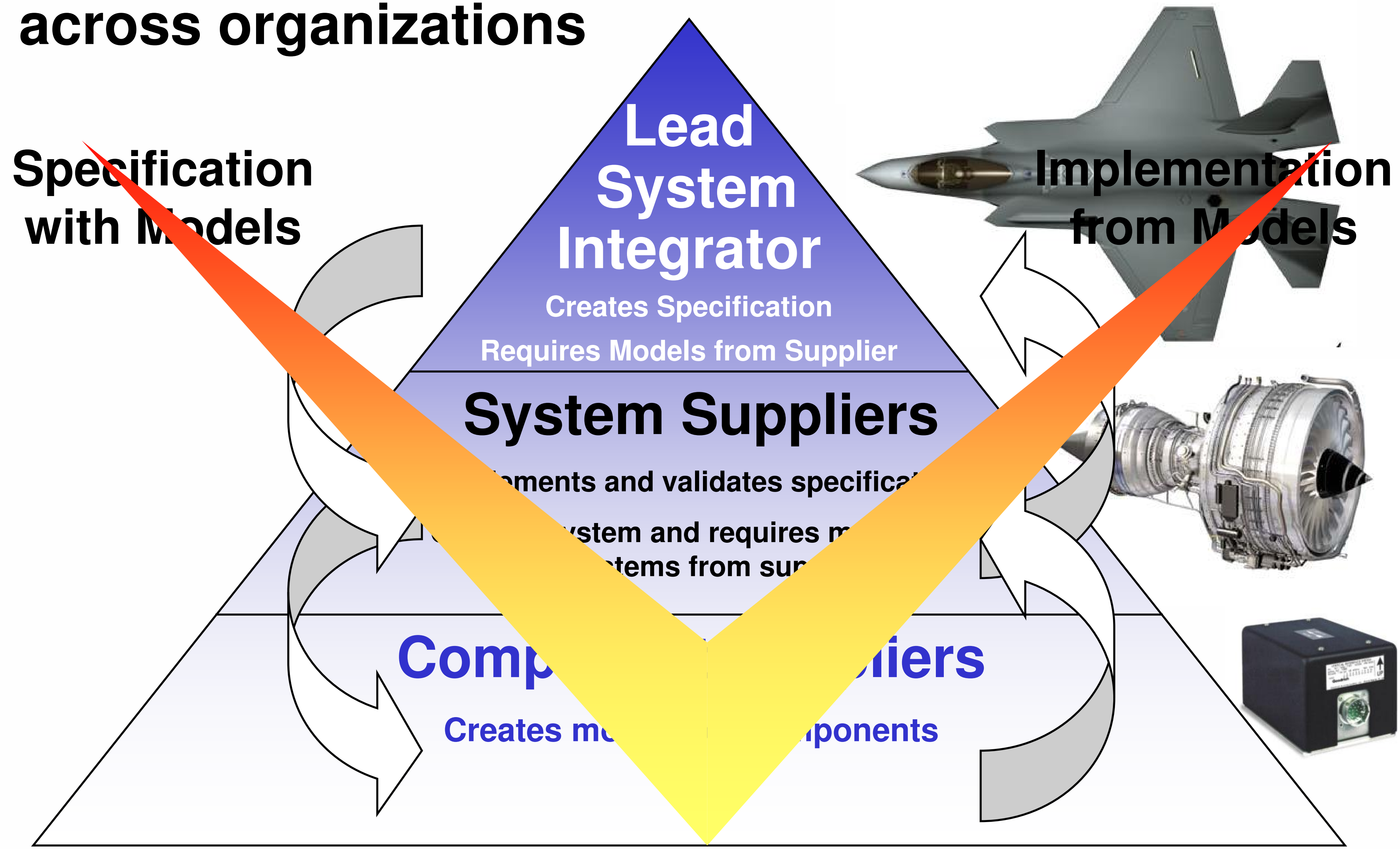
- **All design information is managed and communicated using the model**

  - **Requirements and configuration**
  - **Code specification**
  - **Test cases and results**
  - **Documentation**

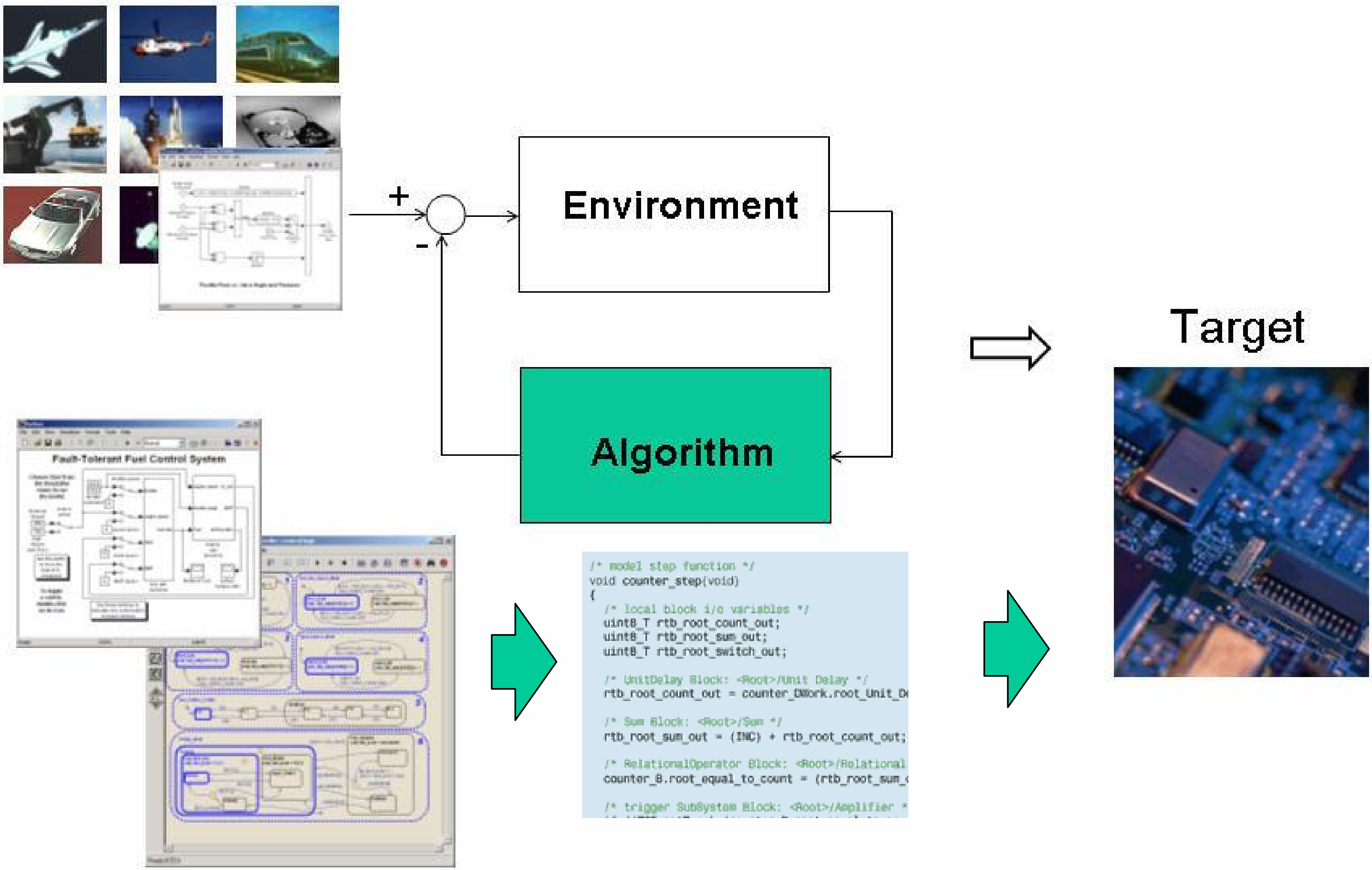
Verification and Validation



Model-Based Design facilitates communication across organizations

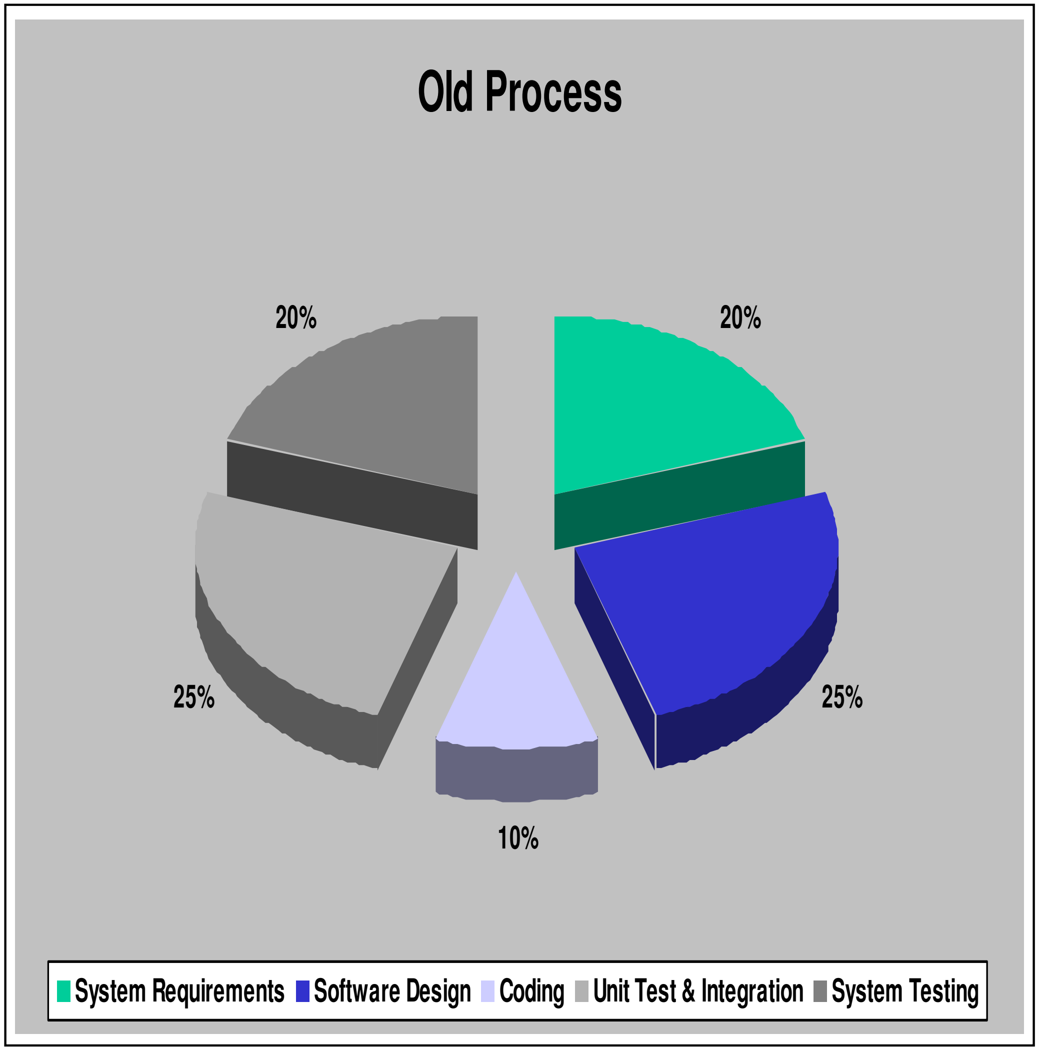


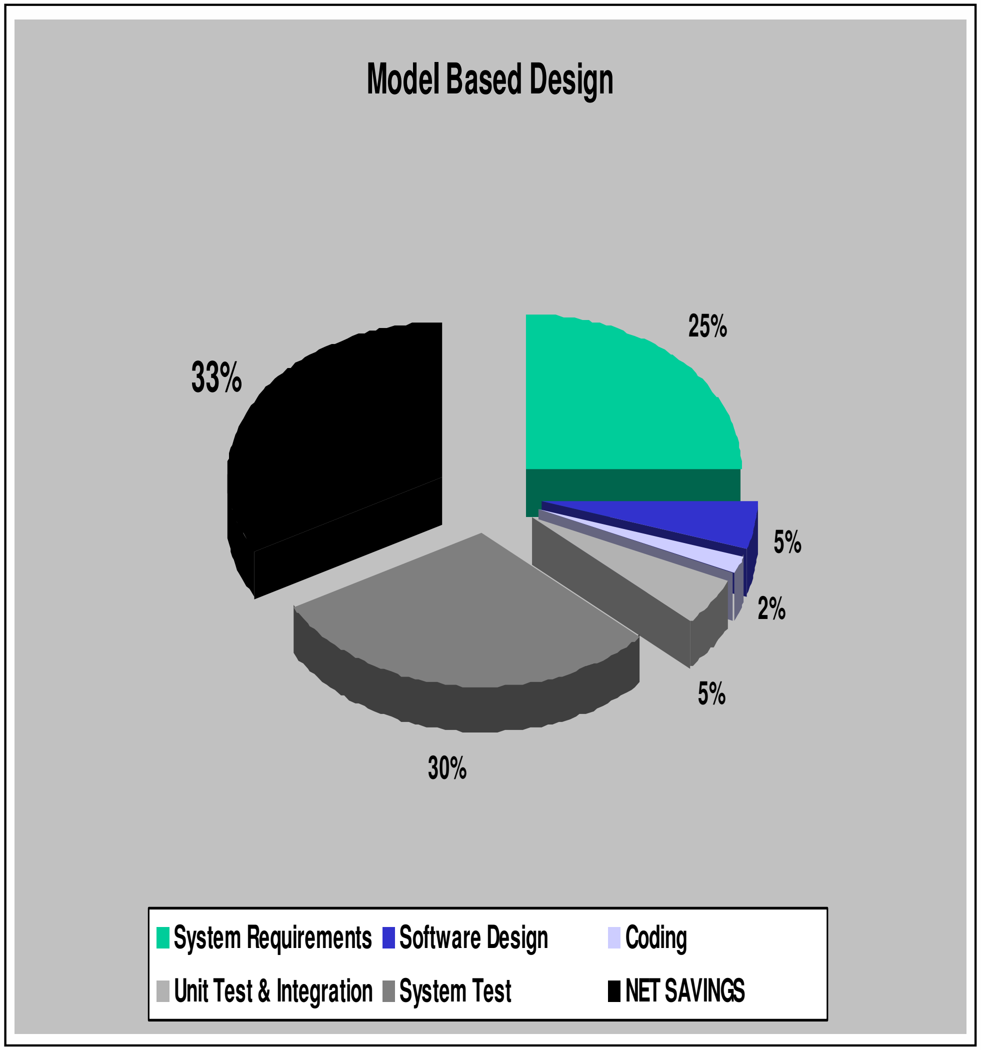
Flight Code Generation Demo



Examples

Large Defense Company (100+ users)





Spend ▲ time and effort on system requirements and testing
Spend ▼ time and effort on detailed design and implementation

Model and Code Re-use

- Gains from Model-Based Design approach are accelerated for:
  - System upgrade engineering and deployment
  - System variant engineering and deployment
  - New design based on existing design

Lockheed Martin

F-35 Joint Strike Fighter (JSF)

- **Background**
  - **The F-35 is considered the single most important new military aircraft to be produced over the next 30 years. The program will deploy 2,593 aircraft for the U.S. Air Force, U.S. Navy, U.S. Marine Corps, U.K. Royal Air Force, and U.K. Royal Navy**
  - **The F-35 family includes a conventional takeoff and landing (CTOL) variant, an aircraft carrier version (CV), and a short takeoff/vertical landing (STOVL) variant, which provide challenging requirements for the flight-control system**
- **Result (Jan '04)**
  - **Lockheed Martin's Chief Test Pilot, Jon Beesley, conducted an unrestricted evaluation flight in the F-35 real-time simulator lasting over two hours, performing a wide range of maneuvers**
  - **He exercised the sophisticated flight control laws designed and generated by Lockheed Martin's JSF Flight Control Application Software team using MATLAB, Simulink, Stateflow, and Real-Time Workshop Embedded Coder**

Press Release (Jan '04)

http://www.mathworks.com/company/pressroom/index.shtml/article/439

BAE SYSTEMS Aerospace Controls

- **Background**
  - **Develop Safety-critical systems using BAE SYSTEMS CsLEOS™ Fault Tolerant RTOS**
  - **Use MathWorks tools for design and flight code generation**
    - **Simulink and Stateflow**
    - **Real-Time Workshop® Embedded Coder**
    - **Straightforward integration of CsLEOS with generated code**
- **Results (April '04)**
  - **CsLEOS and Real-Time Workshop Embedded Coder code are successfully being deployed in:**
    - **Integrated Vehicle Management System Computer used on the Pegasus X-47A unmanned combat aircraft, which was developed by Northrup Grumman**
    - **Flight control computer being designed for the Boeing-built C-17 Globemaster III Air Force transport**
    - **Fly-By-Wire Flight Control For Sikorsky's S-92 AND H-92 Helicopters**
    - **Other current and future systems being developed per RTCA DO-178B Level A and ARINC-653**

Aerospace Engineering (April '04)

Sandia Implements High-Performance Radar Receiver Using MathWorks and Xilinx DSP Design Tools

The Challenge

- To design a digital radar receiver and implement it on the Virtex-II platform FPGA within strict time and budgetary limits

The Solution

- Use MathWorks and Xilinx DSP design software to create, simulate, test, and synthesize designs into hardware

The Results

- Accelerated design
- High-speed simulation
- Accurate representation of the actual system

The first prototype of the digital IF receiver module

"We are so impressed with the tools and the direction in which The MathWorks and Xilinx are going that we plan to make this our mainstream DSP design flow."

Dale Dubbert, Sandia National Laboratories

Rod Millen Special Vehicles Develops Suspension System for Military Vehicle Using MathWorks Tools

The Challenge

- To design an innovative active suspension system for an unmanned ground vehicle (UGV)

The Solution

- Use MathWorks tools to design and optimize the suspension components and active control systems of the UGV

The Results

- Defect detected before production
- Full vehicle hardware prototypes eliminated
- Development time reduced

RMSV-designed Unmanned Combat Ground Vehicle.

"MATLAB, Simulink, and SimMechanics enable us to quickly perform modeling, simulations, and dynamic analysis of next-generation, off-road vehicle suspensions. I consider these tools to be essential for efficient design of advanced control systems. "

Dr. Eric Anderfaas,
Rod Millen Special Vehicles

Major Customers

astrium
AIRBUS
EADS
BOEING
EMBRAER
LOCKHEED MARTIN
BAE SYSTEMS
GENERAL DYNAMICS
Strength on Your Side™
Raytheon
Rolls-Royce
NORTHROP GRUMMAN
Honeywell
THALES
United Technologies
Alenia
GOODRICH
ATK ALLIANT TECHSYSTEMS
MOOG
SAAB
smiths
KONGSBERG
snecma
DRS TECHNOLOGIES
STORK Fokker
L3 communications
Parker
MTU Aero Engines
HARRIS
Ultra ELECTRONICS
Rockwell Collins
LIEBHERR Aerospace
DIEHL Avionik Systeme
ITT Industries Engineered for life

Sandia National Laboratories
THE AEROSPACE CORPORATION
QinetiQ
LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
APL The Johns Hopkins University Applied Physics Laboratory
Los Alamos NATIONAL LABORATORY
DRAPER LABORATORY
MITRE
SAIC An Employee-Owned Company

DGA
NASA
esa
National Defence
DEFENCE Science and Technology Organisation
DSO NATIONAL LABORATORIES
DLR
DEPARTMENT OF DEFENSE UNITED STATES OF AMERICA
SENER
[dstl]

Toyota and DENSO

Mass Production ECUs

- **Background**
  - **“Toyota has a ‘kaizen’ way of always pursuing evolution and innovation, and challenging the improvement,” said Kazuhiko Hayashi, General Manager, Toyota**
  - **We expect that our consistent use of these tools will allow us to accelerate our ability to bring DENSO products to the market more quickly,” said Shinji Shirasaki, Director, DENSO**
- **Results (Dec ‘03)**
  - **Toyota and DENSO will use The MathWorks simulation, modeling, and code generation products in production programs**
  - **Real-Time Workshop Embedded Coder is used to generate, test, and deploy production C code for complex, embedded systems**

Press Release (Dec ’03)

http://www.mathworks.com/company/pressroom/index.shtml/article/438

SAE TECHNICAL PAPER SERIES

2004-01-0269

Multi-Target Modelling for Embedded Software Development for Automotive Applications

Grantley Hodge, Jian Ye and Walt Stuart
Visteon Corporation

Reprinted From: In-Vehicle Networks and Software, Electrical Wiri
and Electronics and Syste

ISBN 0 7680 1435-5

SAE Internat
2004 SAE W
De
Ma

400 Commonwealth Drive, Warrendale, PA 15096-0001 U.S.A. Tel: (724) 776-4841 Fax: (724) 776-5760 W

Table 2 shows ROM and RAM comparisons between hand code and auto code for a floating-point component in some typical powertrain software.

Table 2 ROM and RAM comparison between a floating-point hand code and auto code.

	Hand Code	Auto Code
ROM	6408	6192
RAM	132	112

The auto code has less size of ROM and RAM compared to that of hand code. The auto code is readable and peer reviewed, and checked with the QAC static analysis tool. Most importantly, the auto code is implemented in a real-world powertrain application.

CONCLUSION

A custom data class allowing data type and data scaling information to be incorporated into the model is

Jian Ye, jye5@visteon.com
Walt Stuart, wstuart@visteon.c
Grantley Hodge, ghodge@viste

DEFINITIONS, ACRONYMS

MBDG: Model-Based Develo
Powertrain.

Multi-target Model (Generic contains no data type inform precision).

Target Specific Model: A ge specific data dictionary loade linked to the model.

ROM: Read Only Memory.

RAM: Random Access Memory

KAM: Keep Alive Memory.

MATLAB®: A modeling enviro

SAE Technical Paper 2004-01-0269

Summary

The Value of Model-Based Design

Model-Based Design

- Executable specification
- Design with simulation
- Implementation through code generation
- Continuous test and verification

Time-to-market

- Get it right the first time

Innovation

- Rapid design iterations
- “What-if” studies
- Unique features and differentiators

Quality

- Reduce design errors
- Minimize hand coding errors
- Unambiguous communication internally and externally

Cost

- Reduce expensive physical prototypes
- Reduce re-work
- Reduce testing

Benefits of Model-Based Design

- **Reduced design risk and improved reliability**
  - Models are simulated to clarify their behavior
  - Models are debugged before software is written
  - Executable specifications reduce ambiguity

- **Reduced implementation risk and time**
  - As soon as the model works, equivalent code can be produced automatically with no additional effort or manual coding errors
  - Code for rapid prototyping as well as embedded targets
  - Weeks to months of coding and debugging time are saved

The MathWorks 4th annual

International Aerospace & Defense Conference

October 4-7, 2004, Newton, MA, USA

Monday, October 4th	Tuesday, October 5th	Wednesday-Thursday, October 6-7th
Program Breakouts 2004 **Invitation-only Session*	**Aerospace & Defense Symposium** 2004 *Public, "industry user group"*	**MALC 2004 MathWorks Aerospace Leadership Council** **Natick, MA, USA** **Invitation only, NDA-bound, technical requirements-gathering meeting*
*F-35 JSF *JTRS *Airbus A380/A400M	*Latest Product News *Partner Exhibits *Customer Testimonials	*Automatic Code Generation for Real-Time Systems *Signal Processing and Communication System Design *Testing, Verification and Validation *Large Scale Modeling and Simulation *Desktop Application Deployment *Algorithm Development, Post-Test Data Analysis & Data Exploration *Image Processing and Geospatial Analysis