Tailoring Tools for Application-Oriented Development Using Model-Based Design

Tom Erkkinen
MathWorks
The Origins of MATLAB

By Cleve Moler, MathWorks

In the late 1970s, following Wirth’s methodology, I used Fortran and portions of LINPACK and EISPACK to develop the first version of MATLAB. The only data type was “matrix.” The HELP command listed all of the available functions, with their names abbreviated.

```
ABS  ANS  ATAN  BASE  CHAR  CHOL  CHOP  CLEA  COND  CONJ  COS
DET  DIAG  DIAR  DISP  EDIT  EIG  ELSE  END  EPS  EXEC  EXIT
EXP  EYE  FILE  FLOP  FLPS  FOR  FUN  HESS  HILB  IF  IMAG
INV  KRON  LINE  LOAD  LOG  LONG  LU  MACR  MAG  NORM  ONES
ORTH  PINV  PLOT  POLY  PRIN  PROD  QR  RAND  RANK  RCON  RAT
REAL  RETU  RREF  ROOT  ROUN  SAVE  SCHU  SHOR  SEMI  SIN  SIZE
SQRT  STOP  SUM  SVD  TRIL  TRIU  USER  WHAT  WHIL  WHO  WHY
```

There were only 80 functions. There were no M-files or toolboxes.

The bald and not excessively bald and not excessively smart hamster obeyed a terrified and not excessively terrified hamster.
Early MathWorks Landscape

- Simulink
  - Blocksets (external)
  - Toolboxes (external)

- MATLAB
  - Blocksets
  - Toolboxes
Today’s Landscape
Today’s Landscape – Prolific
Evolving Landscape

User Apps

Blocksets (external)

Toolboxes (external)

Simulink

MATLAB

Toolboxes

Blocksets
Creating Apps
The Adoption of MATLAB Apps and Toolboxes at Jaguar Land Rover

Dave Barry, Jaguar Land Rover

Engineering teams across Jaguar Land Rover rely on MATLAB based apps and toolboxes to develop a wide range of advanced vehicles. This session discusses:

Evolving Landscape

User Apps

Simulink

Toolboxes (external)

MATLAB

Application-Oriented Products

Blocksets (external)

Toolboxes
Application-Oriented Products

Why Application Oriented?

- Basic blocks and functions no longer suffice
- Starting reference point is necessary
- System models are important but hard
- Let your experts focus on their expertise
  - Focus on own component, leverage elsewhere

Cost savings of nearly $2 million per year
Introducing … Products for Application-Oriented Development

Powertrain Blockset

- Model a Powertrain System
  - Use fully assembled reference applications and parameterize the components
  - Learn more

- Design and Test a Controller Model
  - Use built-in controller models, user-defined controller models, and embedded estimators
  - Learn more

- Perform System Design Tradeoff
  - Perform powertrain designs tradeoff studies, such as emissions, fuel economy, and performance
  - Learn more

- Deploy for Hardware-in-the-Loop Testing
  - Perform HIL testing on your controller model
  - Learn more

Automated Driving System Toolbox

- Ground-Truth Labeling
  - Automate labeling of ground truth data, and compare output from the algorithm under test with ground truth data
  - Learn more

- Sensor Fusion and Tracking
  - Perform multi-sensor fusion using multijoint tracking framework with Kalman filters
  - Learn more

- Vision System Design
  - Develop computer vision algorithms for vehicle and pedestrian detection, lane detection, and classification
  - Learn more

- Scenario Generation
  - Generate traffic scenarios and simulate radar and camera sensor outputs to test sensor fusion and control algorithms
  - Learn more
Reference Applications – Starting Point

Powertrain Blockset
Reference Applications – Starting Point

Automated Driving System Toolbox
Workflow Integration - Design

Powertrain Blockset

Automated Driving System Toolbox
Workflow Integration - Code

Powertrain Blockset

Automated Driving System Toolbox

```
% Code Generation for Tracking and Sensor Fusion
function [Vx, Vy] = calculateGroundSpeed(Vxi, Vy, egoSpeed)

% Inputs
% (Vxi, Vy) : relative object speed
% egoSpeed : ego vehicle speed
% Outputs
% [Vx, Vy] : ground object speed

Vx = Vxi + egoSpeed; % calculate longitudinal ground speed
theta = atan2(Vy, Vxi); % calculate heading angle
Vy = Vx * tan(theta); % calculate lateral ground speed
end
```
Workflow Integration - Verification

Powertrain Blockset

Automated Driving System Toolbox
Open and Customizable - Tailoring

Powertrain Blockset
We have other application products

LTE System Toolbox

Overview | Features | Code Examples | Videos | Webinars | What's New | Product Pricing
---|---|---|---|---|---|---

**Design Verification**
Test an LTE design by using a golden reference implementation.
- Learn more

**End-to-End Simulation**
Model and simulate the physical layer of the LTE standard.
- Learn more

**Conformance Testing**
Ensure your designs comply with the supported 3GPP LTE standard releases.
- Learn more

**5G Library**
Simulate 3GPP 5G new radio technologies.
- Learn more
Workflow integration – Your workflow
Your Landscape

The Origins of Your ???

By You

>>Why