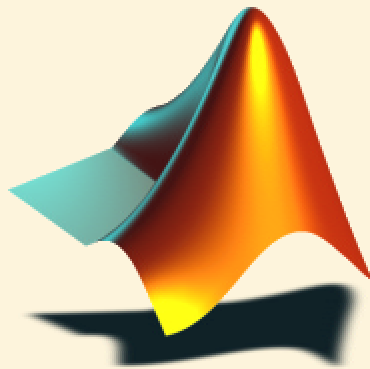


# MATLAB: Platform Architecture

*How the MathWorks puts the “Tower of Power” to work for us.*



**Peter Webb**  
**The MathWorks, Inc.**  
**Natick, MA**

**[pwebb@mathworks.com](mailto:pwebb@mathworks.com)**



# Agenda

---

**The MathWorks at a Glance**

**Overview of MATLAB**

**Technical Computing Market**

**Platform Architecture**

**Organizing for Platforms**



# The MathWorks at a Glance

---

**Founded in 1984, privately held**

**Consistently profitable, with annual double-digit growth since the company's founding**

**1998 yearly revenue reached \$100 million milestone**

**650+ employees today, 200 in product development**

**500,000+ users of our products in 100 countries**



# Worldwide Offices and Distributors

---

- **Headquarters in Natick, MA**
- **Novi, MI Automotive office**
- **Newly-acquired offices in UK, France, Germany, Switzerland, Spain, and Benelux**
- **Pan-European consulting based in UK, France, and Germany**
- **Distributors in 21 countries around the globe**





# MathWorks Product Line

---

**MATLAB (MATrix LABoratory): Matrix calculator**

**Simulink (Simulink): Simulation of physical systems**

**Stateflow: State-based logic diagrams**

**Deployment tools (Compiler, RTW)**

**Domain-specific toolboxes & blocksets**

# The MATLAB Desktop Interface

## Intuitive and easy access to MATLAB features

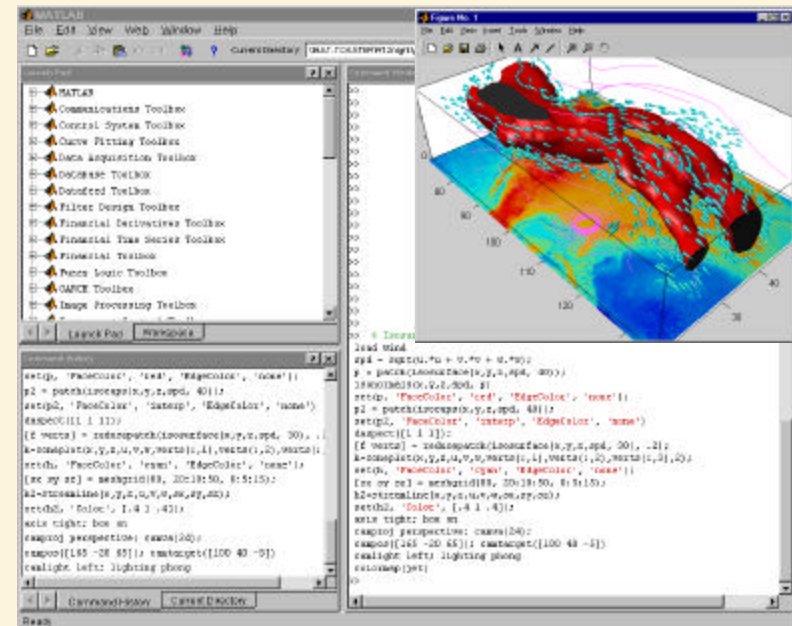
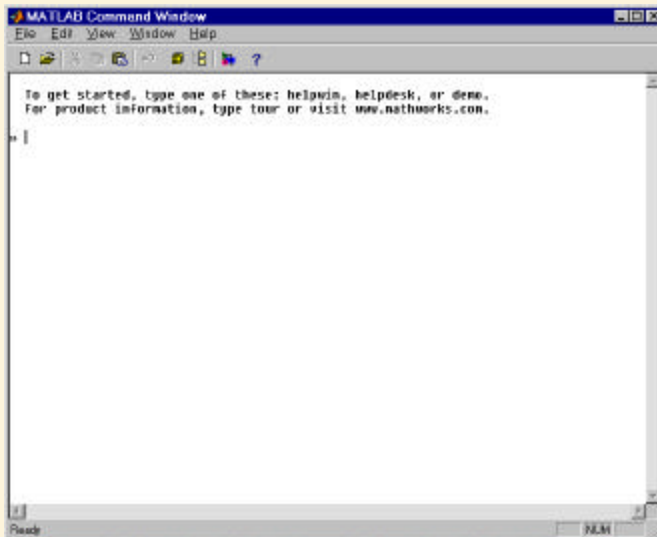
Access to data, code, files, and previous sessions

HTML-based Help and Navigator integrated across products

## Increases speed of analysis and development

## Shortens learning curve

## Customizable to fit user's working model



# MATLAB Desktop Tools

Launch Pad:  
Start  
demos and tools

Desktop Window

Workspace Browser  
lets  
you see your data

Help and  
Help Navigator:  
Find features and  
try them instantly

Workspace Browser,  
M-file Editor,  
Command Window  
and History

Array Editor:  
view and  
edit data

The screenshot shows the MATLAB Desktop Window with several toolboxes open. The Workspace Browser on the left lists variables like 'trans' and 'shad'. The M-file Editor in the center shows a function named 'transparency\_example'. The Command Window at the bottom displays the execution of the function. The History window shows the sequence of commands entered.

The screenshot shows the Workspace Browser window, which displays a list of variables in the current workspace. The variables are organized into columns: Name, Size, Type, and Class. A list of variables including 'trans' and 'shad' is visible.

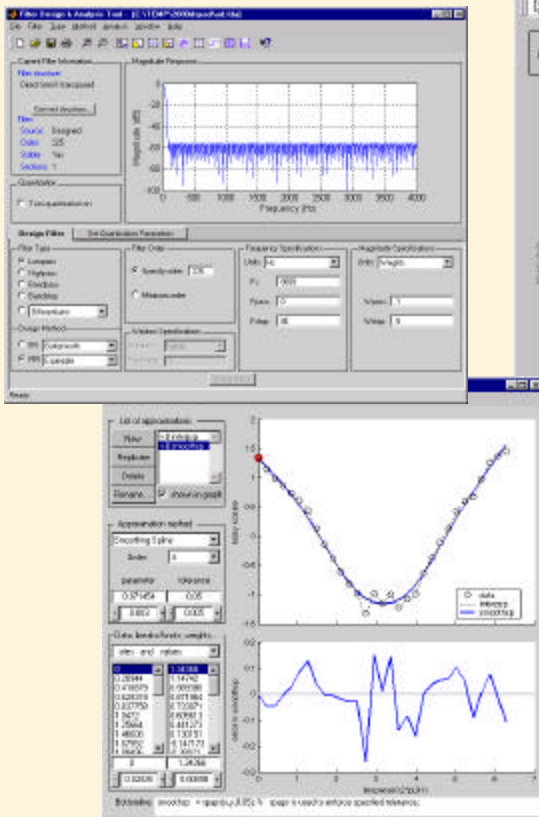
The screenshot shows the MATLAB Help Navigator window. The search term 'transparency' has been entered, and the results are displayed in a list. A 3D surface plot is shown as a preview for the 'Setting a Single Transparency Value for Images' topic.

# MATLAB Toolboxes

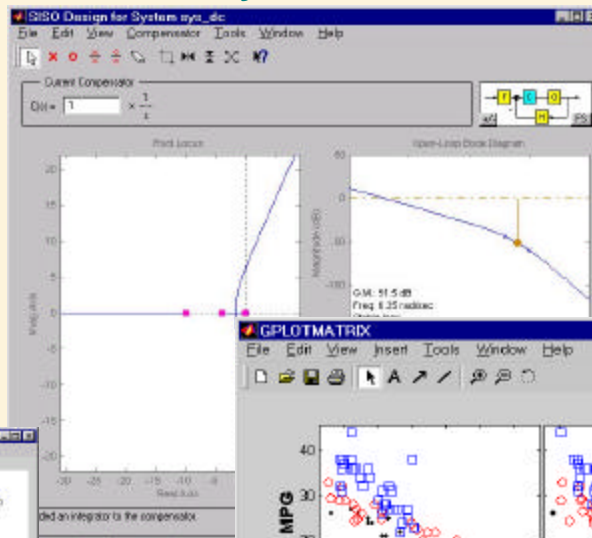
## Easy-to-use interfaces and algorithms

Quickly learn and apply toolbox technology with new interfaces  
 Application-focused examples and demos  
 New algorithms

### Signal Processing

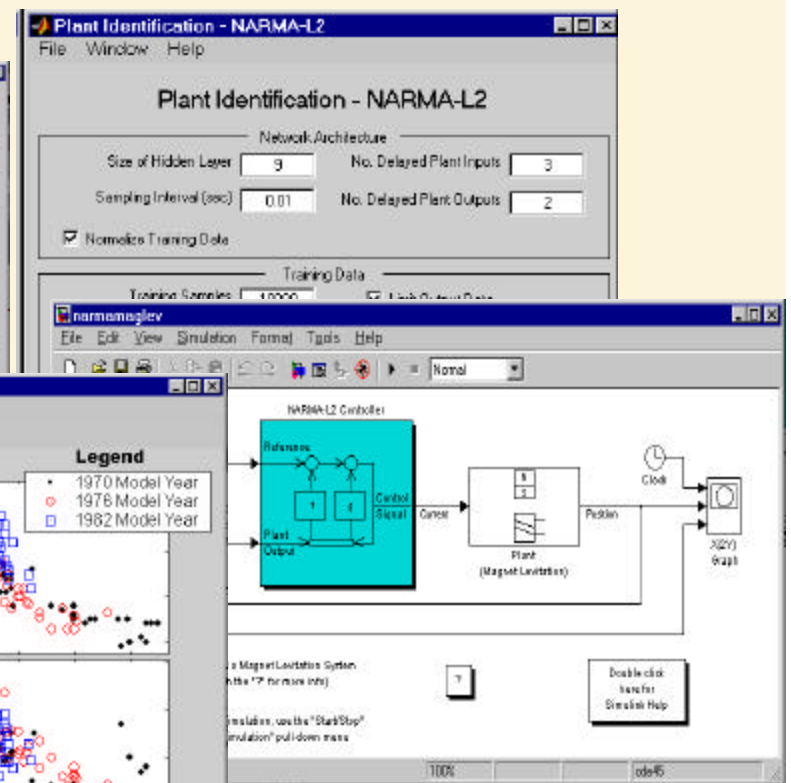


### Control System



### Spline

### Neural Network



### Statistics



# Application Development: Compiler Suite

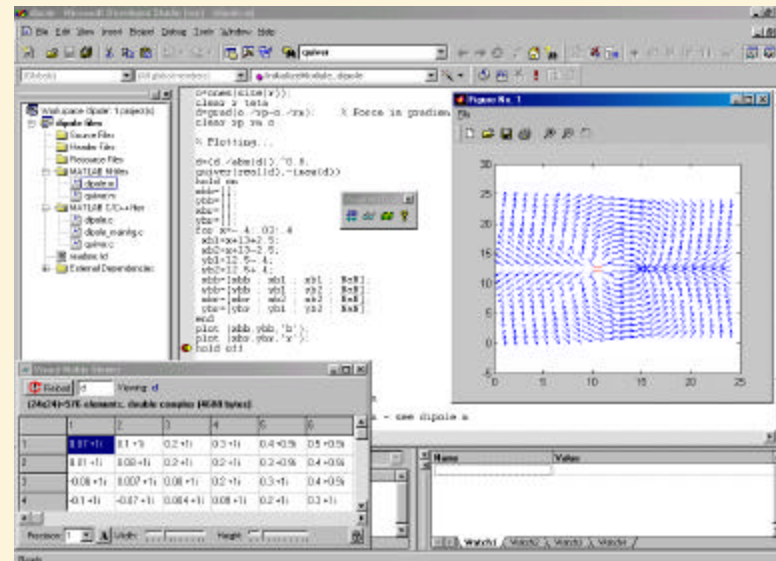
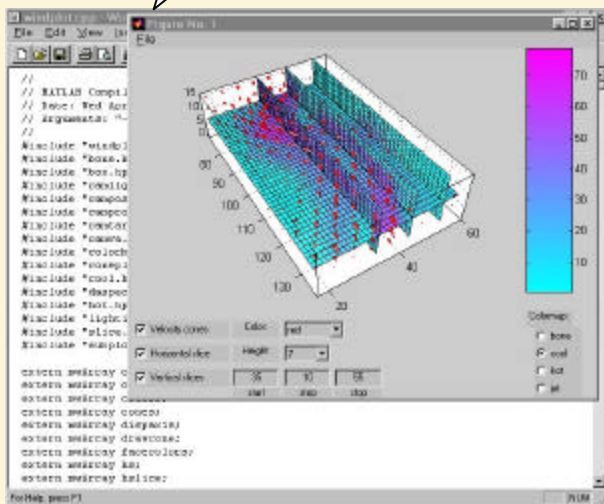
## Converting MATLAB applications to C, C++

Compiles MATLAB 6 applications with math, graphics, GUIs

Integer data types supported (e.g., for image-related tasks)

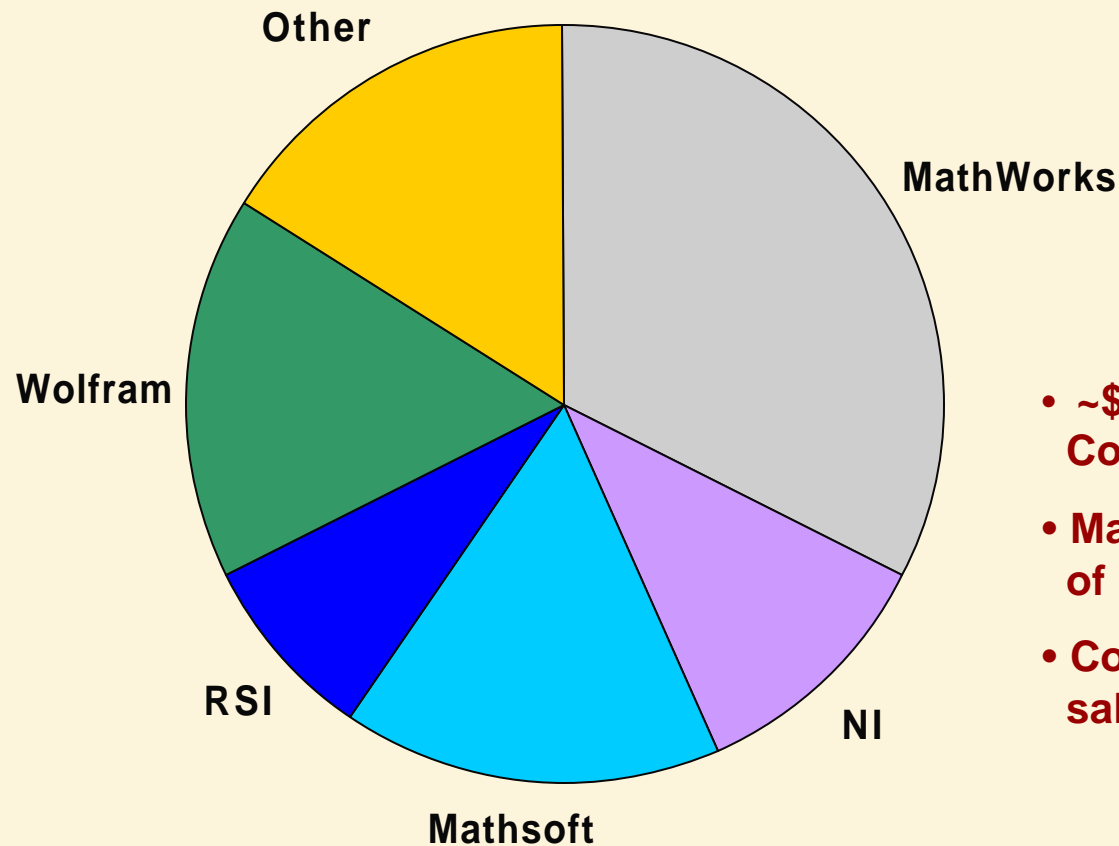
Speed improvements in compiled code vs. original M-files

Generated C code  
and running  
application



Build M-file applications directly from  
Microsoft Visual Studio with the MATLAB  
add-in for Visual Studio

# Technical Computing Market in 1999

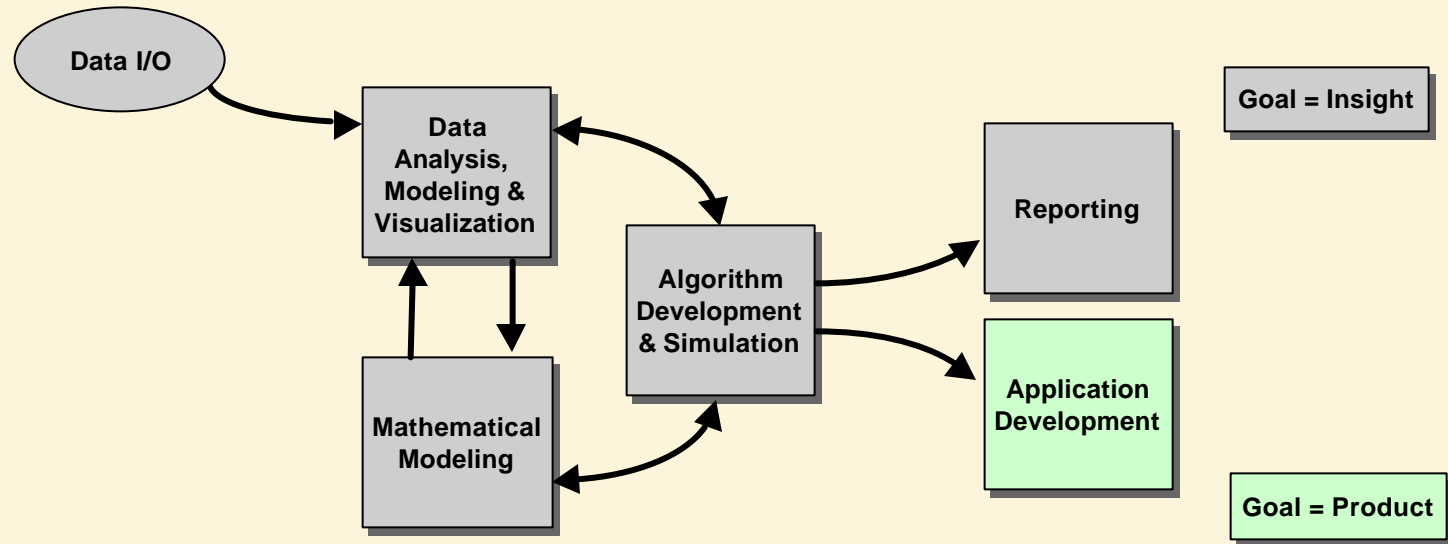


**Revenue (Millions \$)**

- ~\$170 million Technical Computing market in 1999
- MathWorks accounted for 35% of overall market share
- Competitive \$\$ estimates reflect sales of comparable software

Notes:  
 Technical Computing (non-Simulink) software sales  
 MathWorks' best estimate based on public sources

# Typical Technical Computing User Tasks



**Research, Exploration, and Discovery**

**Application Development and Deployment**

*User's Need: Tools that encourage creativity while enabling fast results.*

Task may represent the work of one or more people.

A project can involve both Research and Development tasks.

Examples: Newman Haas, Woods Hole, PostBrake, BioDiscovery



# MATLAB Market Segmentation

	<b>DSP Design</b>	<b>Finance</b>	<b>Test &amp; Measurement</b>	<b>Control Design</b>
<b>Best</b>	Add: Real Time Workshop	Add: Financial Derivatives Toolbox	Add: Wavelet & Statistics Toolboxes	Add: Real Time Workshop, Stateflow Coder
<b>Better</b>	Add: Simulink & DSP Blockset	Add: Financial Toolbox	Add: Signal Processing, Instrument Control Toolboxes	Add: Simulink, Stateflow
<b>Good</b>	MATLAB & Control System Toolbox	MATLAB & Optimization Toolbox	MATLAB & Data Acquisition Toolbox	MATLAB & Control System Toolbox
<b>Shared Subsystem: MATLAB</b>				



# MathWorks: Partial Customer List

---

- Analog Devices
- Boeing
- Daimler Chrysler
- Denso
- Eastman Kodak
- Ericsson
- Ford
- General Motors
- Georgia Tech
- IBM
- Lockheed Martin
- Lucent Technologies
- MIT
- Motorola
- NASA
- Nokia
- Northrop Grumman
- Quantum
- Raytheon
- SAAB Aerospace
- Stanford University
- Toyota
- US Air Force, Navy
- Xerox

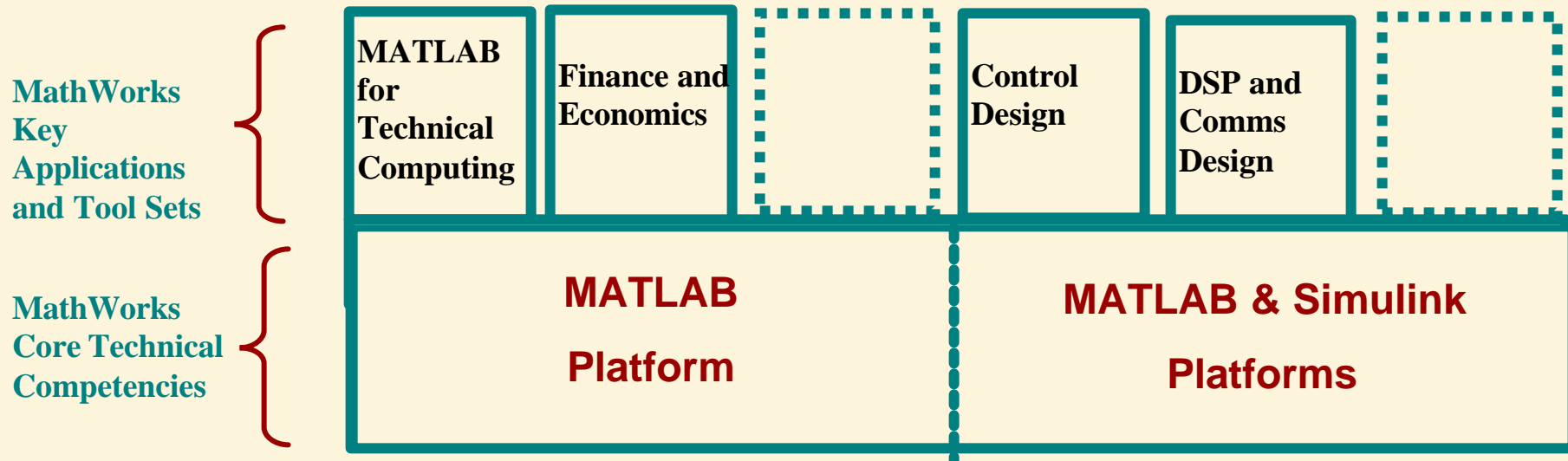
*Related user stories for major accounts available at [www.mathworks.com](http://www.mathworks.com)*



# The MathWorks Business

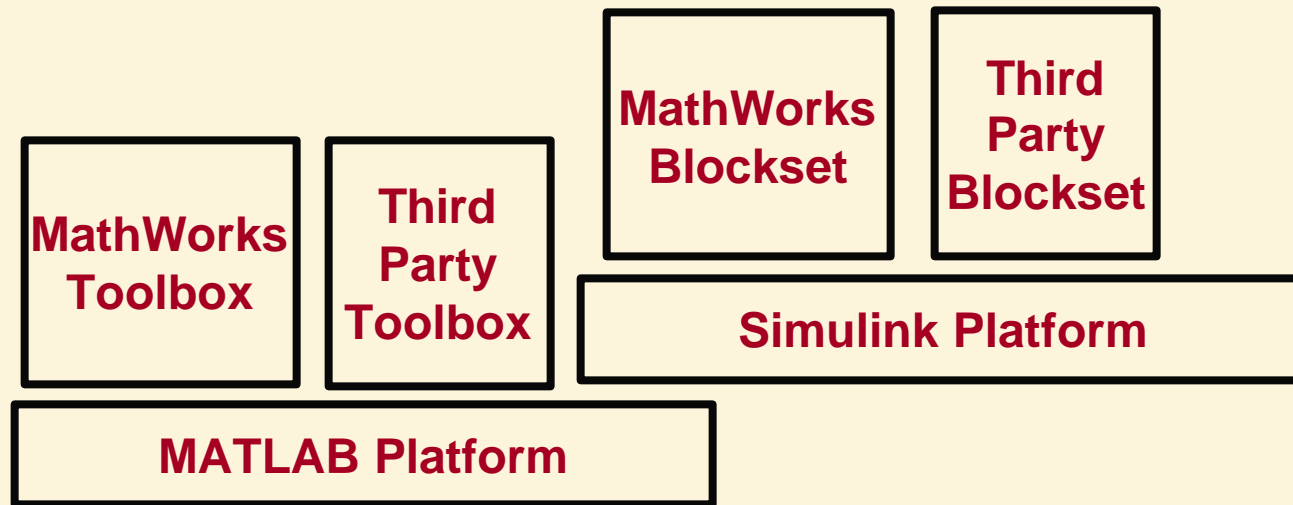
Our mission is to *accelerate innovation, discovery, and development.*

Our tools work across applications, companies, and industries.



## Primary Application Focuses

# Pervasive Platform Architecture



**Platforms, Platforms everywhere**

**MATLAB, Simulink and Toolboxes function as product platforms**

**Common plug in API: the MATLAB Language**



# The MATLAB Platform

---

**MATLAB Language interpreter**

**Interactive environment**

**Common layer of sophisticated matrix math**

**Graphics functions**

**2D plotting**

**3D visualization**

**Application builder tools (GUI builder, etc.)**





# The Simulink Platform

---

**Block diagram environment**

**Block library management**

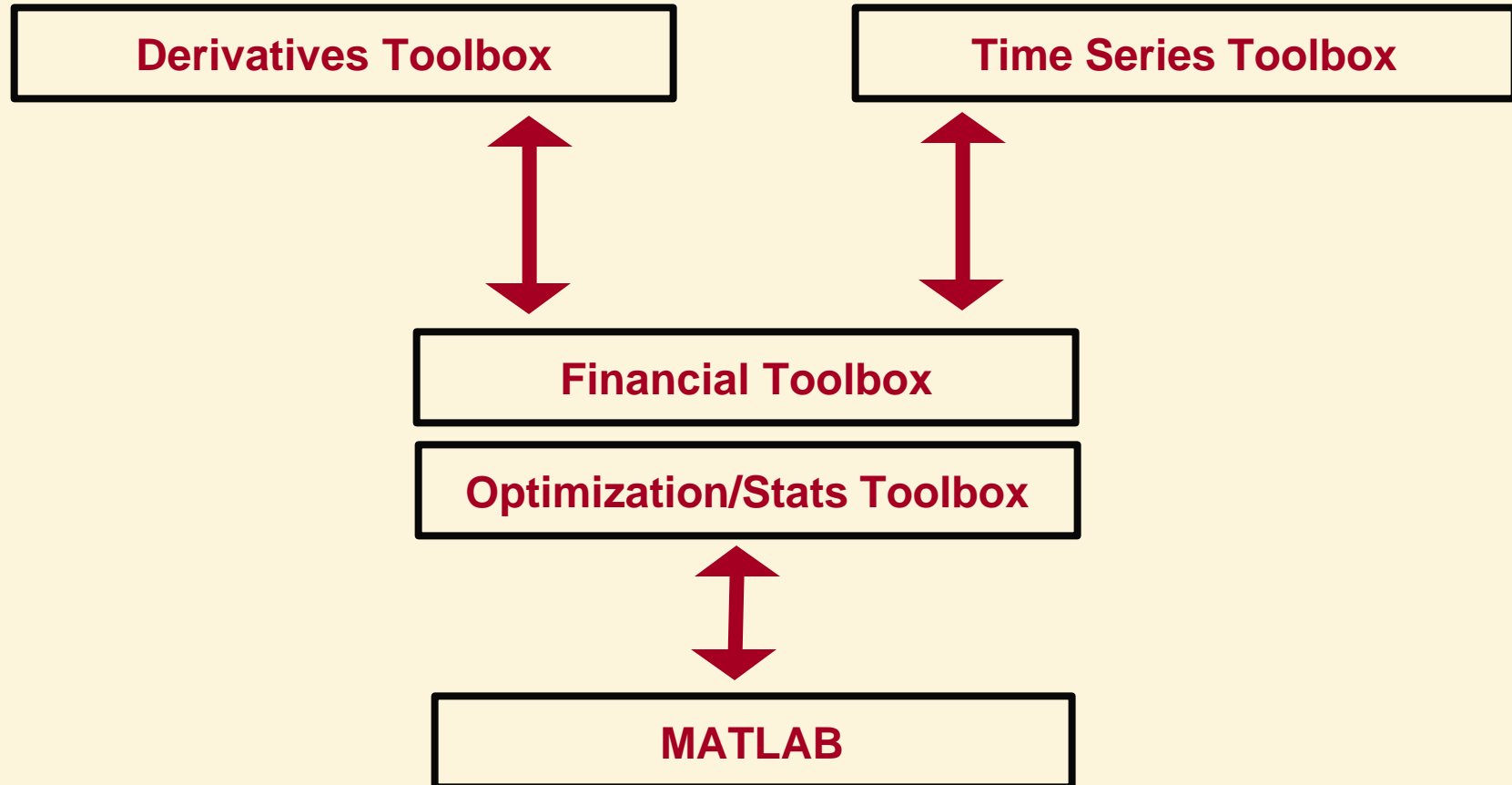
**Common library of basic blocks**

**Integration (mathematical) engine**

**Simulation engine**

**Real-time deployment tools**

# Financial Toolbox: A Toolbox Platform





# The MATLAB Language

---

**C-like syntax, no declarations**

**All variables: matrices (arrays)**

**Turing Complete**

**Object Oriented (allows class definitions)**

**Ability to call other languages (C, Java)**

**Extensible: Add more functions to base API**

# An Example MATLAB Function

```
function H = invhilb(n)
p = n;
H = zeros(n,n);
for i = 1:n
    if i > 1, p = ((n-i+1)*p*(n+i-1))/(i-1)^2; end
    r = p*p;
    H(i,i) = r/(2*i-1);
    for j = i+1:n
        r = -((n-j+1)*r*(n+j-1))/(j-1)^2;
        H(i,j) = r/(i+j-1);
        H(j,i) = r/(i+j-1);
    end
end
end
```



# The Language as a Platform

---

**Ubiquitous**

**Familiar (easy to use)**

**Rich factory API**

**Extensible**

**Themed (matrices)**

**Rich development environment (editor, debugger)**



# Benefits of Platform Architecture

---

*Respond quickly to market needs (new toolbox)*

**Robust infrastructure => rapid development of solution**

**Infrastructure cost broadly amortized (justifies investing in quality)**

**Multiple (value, price) offerings (vary value with price)**

**Open architecture accessible to third parties (industry standard, network effect)**



# Organizing For Platforms

---

**Each platform owned by a cross-functional team**

**Development, doc, QE, marketing -- sit together**

**Team negotiates with mgmt. on strategy**

**Team does not own pricing (but makes recommendations)**

**Sales: every sale includes MATLAB; selling bundles**