MATLAB® & SIMULINK®

- Computation
- Visualization
- Programming
- Modeling
- Simulation
- Implementation

Release Notes for Release 2006a

How to Contact The MathWorks:

`www.mathworks.com`	Web
`comp.soft-sys.matlab`	Newsgroup
`www.mathworks.com/contact_TS.html`	Technical support

`suggest@mathworks.com`	Product enhancement suggestions
`bugs@mathworks.com`	Bug reports
`doc@mathworks.com`	Documentation error reports
`service@mathworks.com`	Order status, license renewals, passcodes
`info@mathworks.com`	Sales, pricing, and general information

508-647-7000 (Phone)

508-647-7001 (Fax)

The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA 01760-2098

For contact information about worldwide offices, see the MathWorks Web site.

Release Notes for Release 2006a

Revision History

March 2006 First printing New for Release 2006a

Contents

Release Notes for R2006a

Release Notes for R2006a

Note The HTML version of the release notes includes many links to other MathWorks documentation and to The MathWorks Web site. If you are reading these release notes in printed or PDF format, we recommend viewing the HTML version if you wish to access more detailed information that is linked to from these release notes.

Introduction

Release 2006a (R2006a) includes many new features and two new products. The major focus of R2006a is on improving the quality of the MathWorks products.

Where to Find Information About This Release

The following describe what's new in this release:

- Release Notes for R2006a (described below)
- Product-specific release notes ("Quick Access to Product-Specific Release Notes" on page 45 includes links to product-specific release notes)

To view important fixed and open bugs in R2006a, you can use the Bug Reports interface on The MathWorks Web site.

Release Notes for R2006a

These general release notes summarize the products updated for R2006a. Major topics are

Highlights of Changes Since R14SP3

These general release notes highlight changes in products since Release 14 with Service Pack 3 (R14SP3). This includes products that updated for R14SP3 and products that have had a Web release since R14SP3.

If you are upgrading from a release earlier than R14SP3, you may want to refer to the Release Notes for Release 14 with Service Pack 3 (on the MathWorks Web site).

Printing the Release Notes for R2006a

If you are reading these general release notes for R2006a online and would like to print them, a PDF version is availabe on The MathWorks Web site.

Summary of Product Updates

The release summary table in this section summarizes the kinds of updates made to each product for R2006a.

R2006a includes three new products that were introduced since R14SP3. See "New Products" on page 27 for an introduction to these new products.

- MATLAB® Builder for .NET
- SimBiology
- SimEvents

Which Products Are Listed

Products are listed in the following sections *only* if they have significant updates for R2006a.

However, the release notes for products not listed in the following tables may discuss upgrade issues or other important information. See "Quick Access to Product-Specific Release Notes" on page 45 for links to all the product-specific release notes.

Explanation of Table Columns

The release summary table that follows has these columns:

Product

MATLAB® and Simulink® are listed first in the table. The rest of the MathWorks products are listed alphabetically.

An asterisk (*) after a product name indicates the product has had a Web release since R14SP3.

Product name links take you to the individual release notes for specific products.

Note The product-specific release notes are available in the online documentation for each product and on the MathWorks Web site.

New Features

The links in the “New Features” column take you to a summary of major new features for the product. For details about a product’s new features, see that product’s release notes.

Major Bug Fixes

A “Yes” in the “Major Bug Fixes” column indicates R2006a includes major bug fixes for the product.

Note To view important fixed and open bugs in R2006a, use the Bug Reports interface on the MathWorks Web site. If you are not already logged in to to your MathWorks Account, you will be prompted to log in or create a MathWorks Account.

To view bug fixes introduced before R14SP2, see the product-specific release notes.

Compatibility Considerations

A “Yes” in the “Compatibility Considerations” column indicates that there are compatibility issues associated with upgrading from R14SP3 to R2006a. The “Yes” link takes you to the product-specific release notes, where the Compatibility Summary table provides more information.

Release Summary

Product (Links to Release Notes)	New Features	Major Bug Fixes	Compatibility Considerations
MATLAB®	Yes	Yes	Yes
Simulink®	Yes	Yes	Yes
Aerospace Blockset	Yes	Yes	No
Bioinformatics Toolbox*	Yes	No	No
Communications Blockset	Yes	Yes	Yes
Communications Toolbox	Yes	Yes	No
Control System Toolbox	Yes	No	
Data Acquisition Toolbox*	No	Yes	No
Datafeed Toolbox	Yes	No	No
Distributed Computing Toolbox*	Yes	Yes	No
Embedded Target for Motorola® MPC555	Yes	Yes	Yes
Embedded Target for TI C2000™ DSP	Yes	Yes	No
Embedded Target for TI C6000™ DSP	Yes	Yes	Yes
Filter Design HDL Coder	Yes	No	Yes
Filter Design Toolbox	Yes	Yes	No
Financial Toolbox	Yes	No	No
Fixed-Point Toolbox	Yes	Yes	Yes

Product (Links to Release Notes)	New Features	Major Bug Fixes	Compatibility Considerations
GARCH Toolbox	Yes	No	No
Gauges Blockset	No	Yes	No
Image Acquisition Toolbox*	Yes	Yes	No
Image Processing Toolbox	Yes	Yes	Yes
Instrument Control Toolbox	Yes	No	No
Link for Code Composer Studio™ Development Tools	Yes	No	Yes
Link for ModelSim®	Yes	Yes	No
Mapping Toolbox	Yes	Yes	Yes
MATLAB® Compiler	Yes	Yes	No
MATLAB® Distributed Computing Engine*	Yes	Yes	No
MATLAB® Report Generator	No	Yes	No
Model-Based Calibration Toolbox*	No	Yes	No
Neural Network Toolbox	Yes	No	Yes
OPC Toolbox	Yes	Yes	No
Optimization Toolbox	No	Yes	No
Partial Differential Equation Toolbox (no release notes)	No	Yes	No

Product (Links to Release Notes)	New Features	Major Bug Fixes	Compatibility Considerations
Real-Time Workshop®	Yes	Yes	Yes
Real-Time Workshop® Embedded Coder	Yes	Yes	Yes
RF Blockset	Yes	Yes	No
RF Toolbox	Yes	Yes	No
Robust Control Toolbox	No	Yes	No
Signal Processing Blockset	Yes	Yes	No
Signal Processing Toolbox	Yes	Yes	Yes
SimDriveline	Yes	Yes	No
SimEvents*	Yes	Yes	Yes
SimMechanics	Yes	Yes	No
SimPowerSystems	Yes	Yes	No
Simulink® Accelerator	No	Yes	No
Simulink® Control Design	Yes	Yes	No
Simulink® Fixed Point	Yes	No	No
Simulink® Report Generator	Yes	Yes	No
Simulink® Response Optimization	Yes	No	No
Simulink® Verification and Validation	No	Yes	No
Spline Toolbox	Yes	Yes	Yes

Product (Links to Release Notes)	New Features	Major Bug Fixes	Compatibility Considerations
Stateflow® and Stateflow® Coder	Yes	Yes	No
Statistics Toolbox	Yes	Yes	No
System Identification Toolbox	Yes	No	Yes
Video and Image Processing Blockset*	Yes	Yes	Yes
Virtual Reality Toolbox	Yes	Yes	No
xPC Target	Yes	Yes	Yes

Summary of R2006a New Features

This section summarizes the major new features and enhancements introduced in R2006a.

MATLAB and Simulink are described first. The rest of the updated MathWorks products are listed alphabetically.

The products summarized below are

You can see a summary of all new functions and models introduced in R2006a in the documentation page on the MathWorks Web site.

MATLAB

- Desktop Tools and Development Environment
  - M-Lint code analyzer, now built into the Editor/Debugger, continuously checks your code for problems and recommends modifications to maximize performance and maintainability.

  - Tab completion for functions and variables in the Editor/Debugger
  - Ability to return to previous points in an M-file in the Editor/Debugger
- Mathematics
  - New `ddesd` function solves initial value problems for delay differential equations with general delays
  - Upgraded optimized Basic Linear Algebra Subprogram (BLAS) libraries for Intel processors on Windows and Linux platforms (MKL 8.0.1) and AMD processors on Linux platforms (ACML 2.7)
  - Improved performance of sparse Cholesky factorization with use of CHOLMOD 1.0 routines
- Programming
  - New regular expression features, including evaluation of MATLAB expressions to perform dynamic matching or replacing of text

Simulink

- Large-scale modeling
  - Simulink now detects and pinpoints excessive zero-crossing events to specific blocks in your model, to help you diagnose and fix performance problems in your model.
  - A new tiled printing feature lets you print a block diagram across multiple pages that can be taped together to create a large printout.
  - The Subsystem block context menu now includes a Convert to Model item that converts an atomic subsystem to a model reference.
  - The new Concatenate block concatenates its input signals to create a single output signal whose elements occupy contiguous locations in memory. The block typically uses less memory than the Matrix Concatenation block that it replaces.
  - Support for disabling the inline parameters option for the top model in a model reference hierarchy. This lets you tune the top model's parameters without sacrificing the simulation speed resulting from model referencing.
  - New command-line API for running the Model Advisor from MATLAB
  - Signal logs now preserve the hierarchy of bus data, which facilitates generation of test vectors for large models.

- Embedded systems design
  - Support for fixed-point values to specify the values of block parameters, which can improve the performance of generated code
  - Support for using signal objects to initialize any signal or state, not just the outputs of blocks that have an initial condition or value parameter. This increases the reliability of code generated from the model.
  - Simulink now checks to ensure that the values of signal and parameter objects fall within the ranges specified by their Minimum and Maximum properties.
- Greater usability:
  - Support for customizing any Simulink or Stateflow editor menu
  - New MATLAB desktop item on the Model Editor's View menu brings the MATLAB desktop to the top of your screen.
  - Double-clicking a Model block opens the model referenced by the block instead of the Model block's parameter dialog box as in previous releases.
  - You can now configure a Logical Operator block to have the same shape as the IEEE symbol for the logical operation that it performs. This lets you tell at a glance what option the block performs.

Aerospace Blockset

- Compatibility with FlightGear v0.9.9 flight simulator
- Documentation improvements

Bioinformatics Toolbox

The following features were added in Version 2.2 of the Bioinformatics Toolbox. Version 2.2 was released via the Web, after R14SP3. No additional new features were added in Version 2.2.1, which is included in R2006a.

- New GUI to interactively view, explore, and make manual modifications to multiple alignments
- Functions to read Agilent Feature Extraction Software files
- Functions to import the Gene Ontology database from the Web into MATLAB and extract a subset of the ontology

Communications Blockset

- Rician Channel block enhanced
- Channel visualization added to Multipath Rician Fading Channel block
- Viterbi Decoder and Convolutional Encoder blocks updated with puncturing
- Demodulator blocks enhanced
- M-PSK and Rectangular QAM blocks enhanced
- Phase/Frequency Offset block enhanced
- Additional C data type support
- DVBS2 demo added
- Warning for obsolete blocks

Communications Toolbox

- `convenc` and `vitdec` updated with puncturing and erasing
- `pamdemod`, `pskdemod`, and `qamdemod` functions enhanced

Control System Toolbox

- Siso Design Tool
  - Multiloop compensator design support
  - Automated compensator tuning using PID, IMC, and LQG methods
  - Support for compensator design in Simulink Control Design and Simulink Response Optimization
- LTI Viewer
  - `lsim` and `initial` commands let you change input data and initial conditions directly in the LTI Viewer.
- LTI Objects
  - Full support for descriptor state-space functions
  - New commands `stepinfo` and `lsiminfo` display time-response charactertistics
  - Full support for I/O delays in MIMO transfer functions
  - New methods for LTI, FRD, and SS objects

Datafeed Toolbox

- The Thomson Datastream data server added

Distributed Computing Toolbox and MATLAB Distributed Computing Engine

The following features were added in Version 2.0 of the Distributed Computing Toolbox and MATLAB Distributed Computing Engine. Version 2.0 was released via the Web, after R14SP3. No additional new features were added in Version 2.0.1, which is included in R2006a.

- Third-party schedulers supported as an alternative to the MathWorks job manager for running jobs and distributing tasks to the cluster
- Parallel jobs in which the tasks running simultaneously on separate machines are able to communicate with each other during their execution
- Support for adding tasks to running jobs when using the MathWorks job manager
- Unicast support or all communications between distributed computing processes.
- Support for remotely starting and stopping MATLAB Distributed Computing Engine processes.
- Enhanced `nodestatus` command supports monitoring all MATLAB Distributed Computing Engine processes

Embedded Target for Motorola MPC555

- TPU emulation mode to download functions to DPTRAM that are not part of the standard ROM mask of TPU functions

Embedded Target for TI C2000 DSP

- Asynchronous interrupt scheduling support for C280x and C281x
- Expanded support for C2000 targets including:
  - Spectrum Digital F2808 eZdsp(tm) DSP board
  - Custom boards based on the C280x and C281x processor families
- Flash memory programming support

- New C280x Chip Support Library including the following blocks:
  - C280x Hardware Interrupt
  - Idle Task
  - C280x ADC
  - C280x ePWM
  - C280x eQEP
  - C280x eCAN Receive
  - C280x eCAN Transmit
  - From Memory
- New demos for the Spectrum Digital F2812 eZdsp(tm) DSP and F2808 eZdsp(tm) DSP boards

Embedded Target for TI C6000 DSP

- Compatibility with Texas Instruments Code Composer Studio 3.1
- Asynchronous scheduler support added to many blocks
- New host communication blocks in the new Host Communication library, including UDP send/receive blocks and byte manipulation blocks
- New DSP/BIOS blocks in DSP/BIOS Library
- Additional DM642 EVM blocks to support IP/Ethernet communications, including specialized UDP send/receive blocks and IP configuration block
- New Audio ADC and DAC blocks for DM642 EVM
- Additional processors supported by the C6000 Custom Target Preferences block
- Added general blocks to use the new asynchronous scheduler, such as Hardware Interrupt, Task, and CPU Timer blocks
- Two new demos that introduce the asynchronous scheduler in Embedded Target for TI C6000 DSP: “Host to Target Communication” and “Audio and Video Loopback.” In addition, most demos now use the new asynchronous scheduler capability

Filter Design HDL Coder

- Extended control over speed versus area tradeoffs in the realization of single-rate direct form FIR filter designs. To achieve the tradeoff, you can specify for generated HDL filter code either a fully parallel architecture or one of several serial architectures.
- Support for code generation for the Delay filter type

Filter Design Toolbox

- Farrow filters
- Cost method
- Fixed-point report (`qreport`)
- Wave digital filters in IIR halfband and IIR multirate designs
- Arbitrary Magnitude and Phase filter design
- IIR multirate designs using allpass polyphase components

Financial Toolbox

- The functionality previously available in the Financial Time Series Toolbox has been incorporated into the Financial Toolbox.
- The Financial Time Series frequency conversion functions extensively modified and expanded
- New statistical functions have been added:
  - Multivariate normal regression without missing data
  - Multivariate normal regression with missing data (expectation conditional maximization)
  - Least squares regression with missing data (expectation conditional maximization)
- New financial time series data transformation function

Fixed-Point Toolbox

- Slope Bias math support added
- Scaled double data type support added to `fi` object
- Global DataTypeOverride property added to `fipref` object
- Embedded MATLAB support added to more Fixed-Point Toolbox functions
- “round” value added to `fimath` Object RoundMode property

GARCH Toolbox

- Augmented Dickey-Fuller and Phillips-Perron unit root tests on univariate financial time series

Image Acquisition Toolbox

- Support for additional Coreco boards
- Support for Coreco Sapera driver. The Image Acquisition Toolbox now supports both the Coreco IFC driver and the Coreco Sapera driver.
- Support for additional Matrox boards
- Support for QImaging devices
- Native Bayer demosaicing — New color space setting interpolates Bayer pattern encoded images into standard RGB images

Image Processing Toolbox

- Enhanced ICC profile capabilities
  - `iccread` and `iccwrite` functions support recent changes to the ICC specification.
  - `iccread`, `makecform`, and `applycform` functions support DeviceLink, ColorSpace, Abstract, and Grayscale profiles.
  - `iccread` can read parametric curve types.
- New pointer management functions
  - New `iptPointerManager`, `iptGetPointerBehavior`, and `iptSetPointerBehavior` utility functions let you control what happens when a figure's mouse pointer moves over and then exits an object

- New constraint creation function
  - New `makeConstrainToRectFcn` utility function lets you specify drag constraints for the `imdistline`, `imline`, `impoint`, and `imrect` functions. Drag constraints limit how a user can move line, point, and rectangle objects in a figure.

Instrument Control Toolbox

- Support for generic instrument drivers to communicate with arbitrary devices that do not use industry-standard drivers
- Support for LeCroy instrument drivers
- Simulations of the Agilent 33120a function generator and Agilent e3648 DC power supply

Link for Code Composer Studio Development Tools

- Compatible with Texas Instruments Code Composer Studio 3.1.

Link for ModelSim

- Support for a native interface to Verilog models under simulation in ModelSim; VHDL wrappers are no longer required for cosimulation of Verilog code
- Frame-based processing supported in cosimulation
- HDL Cosimulation block supports auto-configuration of ports by sending a query to ModelSim

Mapping Toolbox

- GeoTIFF and PROJ.4 library upgrades
- SDTS++ library upgraded to Version 1.5
- Three new utility functions for NaN-separated polygons and lines:
  - `closePolygonParts` — Close all rings in multipart polygon.
  - `isShapeMultipart` — True if polygon or line has multiple parts.
  - `removeExtraNanSeparators` — Clean up NaN separators in polygons and lines.

- 32-bit floating-point GeoTIFF
- The coastlines in `coast.MAT` now follow the convention used by `geoshow`, `mapshow`, and `mapview` to display polygons with "holes." (Outer contours run clockwise, inner contours run counterclockwise.)
- Three new demos:
  - Converting Coastline Data (GSHHS) to Shapefile Format
  - Plotting a 3-D Dome as a Mesh Over a Globe
  - Unprojecting a Digital Elevation Model (DEM)

MATLAB Compiler

- The MATLAB Compiler supports a new target, .NET components. With the optional MATLAB Builder for .NET product, you can create both COM components that can be used in native code applications and .NET components that can be used in managed code applications.

Neural Network Toolbox

- New dynamic learning topologies
  - Time Delay Neural Network
  - Nonlinear Autoregressive Network (NARX)
  - Layer-Recurrent Network
  - Backpropagation support for arbitrarily connected custom dynamic networks, (both forward-propagation and backpropagation, gradients and Jacobians)
- New quick-start wizard
  - NFTOOL - Neural Fitting Tool
- Networks and Data
  - More general transfer, net input, and weight functions
  - Support for unknown weights and unknown/don't care targets
  - Support for dividing data for training, validating, testing
  - New and unified pre/post processing functions

OPC Toolbox

- `opcregister` function enhanced with a `-silent` option to install OPC Foundation Core components without dialog boxes

Real-Time Workshop

- New application programming interface (API) for managing model build information
- New mechanism for customizing post-code generation phase of the build process
- New model configuration option for suppressing makefile generation
- New RSim target option for feeding Inport blocks with MAT-file data
- Switch block optimization for wide control port signals
- Multiport Switch block enhancement that generates a default switch case statement
- Support for Simulink signal object initialization
- Support for Simulink parameters of all numeric data types, including fixed-point, user-defined, and alias data types
- Support for separate value and data type properties for Simulink parameter objects
- Support for new “simplest” rounding mode for fixed-point Simulink blocks

Real-Time Workshop Embedded Coder

- New nonvirtual subsystem option for generating modular functions
- New Export Function capability for generating software components for function-call subsystems
- New Symbol Format options for precise control of auto-generated identifiers
- New Memory Section options for target-specific data placement
- New API for registering module packaging options, including build process hooks and data object customizations
- New Data Object Wizard M-API for using root I/O block names on signals and enforcing signal object resolution
- Enhanced MPT signal object initialization

RF Blockset

- Upper and lower power limits (between which AM/PM conversion applies) added to Amplifier block in the Mathematical Sublibrary.

RF Toolbox

- `extract` function added
- S-parameter conversion functions added

Signal Processing Blockset

- Integration of filter blocks with Signal Processing Toolbox `fvtool` and `filter` objects
- Transposed direct-form structure added to FIR Decimation block
- Data type specification modes added to CIC Decimation block
- `taylorwin` window type added to Window Function block
- Reduced simulation memory footprint for fixed-point capable blocks
- Improved usability for To Wave Device block

Signal Processing Toolbox

- Taylor window function (`taylorwin`) added
- SPTool Filter Designer replaced by FDATool
- `sgolay` example improved
- `zeroflag` parameter added to `zp2sos`
- Help for objects changed

SimDriveline

- Run-time parameters now enabled for key blocks
- Improved implementation for clutch locking and unlocking
- Model Reference code generation now compatible with variable step solvers

SimEvents

SimEvents was introduced as a new product via Web, after R14SP3. Version 1.1, which is included in R2006a, includes the following new feature:

- Support for partial replication in the Replicate block

SimMechanics

- New block dialog boxes now compatible with Simulink Model Explorer
- Model Reference code generation now compatible with variable step solvers
- Improved CAD translation
- SolidWorks CAD translator now compatible with SolidWorks versions 2004–2006

SimPowerSystems

- Average value models implemented for two more blocks in Electric Drives library
- Transformer blocks with SI units now available
- Open Circuit option added for the RLC blocks
- Several new demos and enhancements to existing demos

Simulink Control Design

- Design compensators directly in a Simulink model. Simulink Control Design automatically analyzes the model and identifies open and closed loop systems with which to populate an enhanced version of the Control System Toolbox SISO Design Tool.
- Tune multiloop compensators in a single environment that allows the effects of coupling between feedback loops to be immediately understood.
- Tune blocks using both block parameters (for example, PID gains) or zero-pole-gain representations.
- Tune compensators in a control system using all of the graphical and automated design tools associated with the Control System Toolbox SISO Design Tool and Simulink Response Optimization.

Simulink Fixed Point

- New “simplest” rounding mode for more efficient code

Simulink Report Generator

- Support for exporting a Simulink model or a Stateflow diagram as a Web View, a rendition of the model or diagram that you can view and navigate in a Web browser that supports Scalable Vector Graphics (SVG)

Simulink Response Optimization

- Perform response optimization within the SISO Design Tool.
- Add frequency-domain design requirements to SISO Design Tool plots.
- Add time-domain design requirements to LTI Viewer plots (when launched from the SISO Design Tool).
- Use response optimization to design compensators for control systems.
  - Tune parameters within compensator blocks (e.g., poles, zeros, gains).
  - Tune open and closed loop systems.

Spline Toolbox

- Controlled extrapolation beyond the basic interval
- Interpolation of planar tangent-continuous piecewise biarc curve

Stateflow and Stateflow Coder

- Embedded system design
  - Stateflow provides a new chart option that lets you initialize the value of outputs every time a chart wakes up, not only at time 0. The option ensures that ouputs are defined in every chart execution and prevents latching of outputs.
- Greater usability
  - Support for customizing any Stateflow Editor menu by adding items and submenus, and by disabling or hiding menu items

Statistics Toolbox

- New parametric methods for copulas including `copulacdf`, `copulapdf`, `copulaparam`, `copularnd`, and `copulastat` functions
- New Markov chain sampler algorithms — Metropolis-Hasting (`mhsample`) and slice sampling (`slicesample`) random number generators
- New random number generators for the Pearson (`pearsrnd`) and Johnson (`johnsrnd`) systems of distributions
- New multivariate normal (`mvncdf`) and t (`mvtcdf`) cumulative distribution functions
- New Bootstrap confidence interval functionality (`bootci`)
- New functionality for jackknife estimation (`jackknife`)
- New Durbin-Watson test functionality (`dwtest`)
- New functionality for finding fractional factorial designs (`fracfactgen`)
- Enhanced anovan support for nested factors and continuous factors
- Enhanced time series data support for `xbarplot`, `schart`, and `ewmaplot` functions
- Robust fitting enhancements for `nlinfit`, `nlparci`, `nlpredci`, and `robustfit` functions
- D-optimal design enhancements for `daugment`, `dcovary`, `rowexch`, `candexch`, `candgen`, `cordexch`, and `x2fx` functions
- New and updated demos

System Identification Toolbox

- `balred` function introduced for model reduction
- Search direction for minimizing criteria can be computed by adaptive Gauss-Newton method
- Maximum number of bisections used by line search increased

Video and Image Processing Blockset

- `Autothreshold` block enhanced
- `Draw Shapes` block enhanced
- `isfilterseparable` function added
- Lane Departure Warning System demo added
- MPlay GUI access changed

Virtual Reality Toolbox

- Utilities library added
- VRLM animation playback controls

xPC Target

- Uploading xPC Target signals to Simulink scopes
- Support for uploading real-time data to Simulink scopes when using Simulink external mode
- Support for the MATLAB Compiler: you can generate redistributable, stand-alone xPC Target applications from your M-files with the MATLAB Compiler
- Support added for a number of boards, including the MPL PATI (PowerPC controlled Analog and Timing I/O Intelligence)

New Products

Three new products have been introduced since R14SP3:

MATLAB Builder for .NET	Create MATLAB based .NET and COM components
SimBiology	Model, design, and simulate biochemical pathways
SimEvents	Model and simulate discrete-event systems

MATLAB Builder for .NET

MATLAB Builder for .NET (also called .NET Builder) is an extension to the MATLAB Compiler. Use MATLAB Builder for .NET to

- Package MATLAB functions so that .NET programmers can access them from C#, VB.NET and C++ with managed extensions, or any CLS-compliant language.
- Create components that preserve the flexibility of MATLAB.

  .NET Builder provides robust data conversion, indexing, and array formatting capabilities. To support the MATLAB data types, .NET Builder provides the MWArray class hierarchy, which is defined in the .NET Builder MWArray assembly. You need to reference this assembly in your managed application to convert native arrays to MATLAB arrays and vice versa.
- Handle errors originating from MATLAB functions as standard managed exceptions.

  Error descriptions contain specific references to the MATLAB code, thus simplifying the debugging process.
- Create COM components.

  COM stands for Component Object Model, which is a software architecture developed by Microsoft to build component-based applications. COM objects expose interfaces that allow applications and other components to access the features of the objects. COM objects are accessible through Visual Basic, C++, or any language that supports COM objects.

Note As of R2006a, the functionality previously available in MATLAB Builder for COM has been incorporated into MATLAB Builder for .NET. Customers current on maintenance for MATLAB Builder for COM will receive a license for MATLAB Builder for .NET.

SimBiology

SimBiology is an integrated environment for modeling biological processes, simulating the dynamic behavior of these processes, and analyzing simulation and experimental data. These features were included in Version 1.0 of SimBiology, which was released in Web-downloadable form after R14SP3. There are no additional new features introduced in R2006a.

- Graphical user interface — SimBiology provides access to the command-line functionality through a graphical user interface (GUI). All the features in SimBiology are also accessible and executable from the MATLAB Command Window.
- Data formats and projects — Organize and save related models, simulation data, and analysis results in project files. Save user-defined kinetic laws and units. Share models by exporting SBML level 2 files.
- Modeling — Create biological models by adding components that include reactions, species, parameters, kinetic laws, rules, and submodels.
- Simulation — Select either deterministic solvers with dimensional analysis, or stochastic solvers with dimensional analysis and unit conversion.
- Analysis — SimBiology is fully integrated with MATLAB. Log data during a simulation and analyze results in MATLAB.

SimEvents

SimEvents extends Simulink with tools for modeling and simulating discrete-event systems using queues and servers.

With SimEvents you can create a discrete-event simulation model in Simulink to simulate the passing of entities through a network of queues, servers, gates, and switches based on events. SimEvents and Simulink provide an integrated environment for modeling hybrid dynamic systems containing continuous-time, discrete-time, and discrete-event components.

MathWorks Products in Transition in R2006a

In R2006a, the following two products have been incorporated into other MathWorks products.

MATLAB Builder for COM

As of R2006a, the functionality previously available in MATLAB Builder for COM has been incorporated into MATLAB Builder for .NET. The MATLAB Builder for .NET documentation contains information previously included in the documentation for MATLAB Builder for COM.

Customers who are current on maintenance for MATLAB Builder for COM will receive a license for MATLAB Builder for .NET.

Financial Time Series Toolbox

As of R2006a, the functionality previously included in the Financial Time Series Toolbox is now included in the Financial Toolbox.

Customers who are current on maintenance for the Financial Time Series Toolbox and the Financial Toolbox will continue to have access to this functionality via the Financial Toolbox.

Upgrading from a Previous Release

These topics summarize the potential compatibility considerations when upgrading from R14SP3 to R2006a:

Compatibility Considerations in This Release

Upgrading to R2006a is not expected to cause any significant compatibility problems for most users.

To see if any known compatibility considerations exist for your product, see "Release Summary" on page 6. If the table shows that there are compatibility considerations for any of your products, follow the link to the product's release notes and refer to the table summary of compatibility considerations. From this table you can link to details. If you are upgrading from a release before R14SP3, refer to entries in the table that summarize compatibility considerations with earlier versions.

The compatibility considerations likely to be noticed by the broadest cross-section of users are summarized in the following table. For more information, see the product release notes:

Product and Area	Compatibility Consideration	Potential Impact	Recommended Actions
MATLAB External Interfaces	MEX-files and utilities that rely on the GCC compiler might need to be recompiled due to an upgrade to GCC. 3.4.	At the source-code level, GCC 3.4 now complies much more strictly with the C++ standard and rejects certain constructs that were previously accepted, but which are contrary to the language standard.	This means that all objects linked into MATLAB (either statically or dynamically) must be recompiled with GCC 3.4.
Simulink	Certain MEX-files created in versions before Version 6 (R14) will need to be recompiled to work in subsequent versions.	S-functions created prior to R14 that access the sigmap, siglists and sigregions structures might generate segmentation violations.	Recompile S-functions in Version 6 (R14) or a more recent version.
Real-Time Workshop	`ComplexSignal`, `DataTypeIdx`, `Dimensions`, and `Width` fields removed from records. `BlockOutputs`, `Dworks`, `ExternalInputs`, `ExternalOutputs`, and `Parameters` in *`model`*`.rtw`.	If your application retrieves data from *`model`*`.rtw` without using documented TLC library functions, it will produce incorrect results.	Reprogram your application to use the documented TLC library functions to retrieve data from *`model`*`.rtw`.

Compatibility Considerations for New Functions and Models

The introduction of new functions and models could cause a conflict between any of your own M-files, models, and variables having the same names.

Example

If you created a function named `idivide` in a previous release, it might conflict with the new MATLAB `idivide` function (introduced in R2006a). Another conflict that might arise is when you load a MAT-file that contains a variable named `idivide`.

Recommendations

- In the By Product List of new functions and models (on the documentation page of the MathWorks Web site), scan the products you have installed for function, model, and variable names you already use, and note potential conflicts.
- Because you could experience conflicts with names from products you do not use, scan the Alphabetical List of new functions and models for all products (on the documentation page of the MathWorks Web site), and note any other potential conflicts. For example, you might have a name conflict if you share files with users who have different products installed.
- To identify and address name conflicts, see these topics in the MATLAB Programming documentation:
  - Avoid Using Function Names for Variables
  - Naming a Function Uniquely

System Requirements

Note For more information on system requirements, visit `http://www.mathworks.com/support/sysreq/r2006a/`

All Platforms

- CD-ROM or DVD drive (for installation)
- 512 MB RAM (1024 MB recommended)
- 500 MB disk space

Microsoft® Windows®

32-Bit Operating Systems

- Windows XP(Service Pack 1 or 2); Windows 2000 (Service Pack 3 or 4); or Windows Server 2003
- Intel® Pentium® III processor or later, and AMD processors
- 16-, 24-, or 32-bit OpenGL® capable graphics adaptor

Note R2006a does not support Windows NT®; R14SP2 (Release 14 with Service Pack 2) was the last release to support Windows NT.

64-Bit Operating Systems

- Windows XP x64
- Intel® EM64T or AMD AMD64 processors
- 16-, 24-, or 32-bit OpenGL® capable graphics adaptor

Matrix Size Limitations. Historically, MATLAB matrices have been limited in size to those that would fit in a 32-bit address space. Some of those limitations have been lifted, but some remain. Specifically, the data type used to index into an

mxArray is still a 32-bit signed integer. This limits the number of elements in any one array to INT_MAX-1, or 2147483646 (approximately 2*10^9). With this limit, you can create matrices up to 16 GB (for doubles). You can create as many of these as your machine has memory to support.

Mac OS® X

32-Bit Operating Systems

- PowerPC G4 or G5 processor
- Mac OS X 10.4, 10.4.2 (Tiger™), or 10.3.8 or 10.3.9 (Panther™); 10.3.9 requires an Apple® Java patch
- 16-bit graphics or higher adaptor and display (24-bit recommended)
- X11 (X Server)

Linux

32-Bit Operating Systems

- Linux built using Kernel 2.4.x or 2.6.x and glibc (glibc6) 2.3.2 or later
- Intel® Pentium® III processor or later, and AMD processors
- 16-bit graphics or higher adaptor and display (24-bit recommended)

64-Bit Operating Systems

- Linux built using Kernel 2.4.x or 2.6.x and glibc (glibc6) 2.3.4 or later
- Intel® EM64T or AMD AMD64 processors
- 16-bit graphics or higher adaptor and display (24-bit recommended)

Matrix Size Limitations. Historically, MATLAB matrices have been limited in size to those that would fit in a 32-bit address space. Some of those limitations have been lifted, but some remain. Specifically, the data type used to index into an mxArray is still a 32-bit signed integer. This limits the number of elements in any one array to INT_MAX-1, or 2147483646 (approximately 2*10^9). With this limit, you can create matrices up to 16 GB (for doubles). You can create as many of these as your machine has memory to support.

CDF Writing Not Supported. CDF writing is not supported, but you can read CDF files.

UNIX (Solaris)

Note R2006a does not support HP-UX; R14SP3 (Release 14 with Service Pack 3) was the last release to support HP-UX.

32-Bit Operating Systems

- Sun® Solaris® 10, 9, or 8 (running R2006a on Solaris 8 or 9 requires that the operating system have the current recommended patches; see Solution 1-21Q99R)
- SPARC or ultraSPARC
- 16-bit graphics or higher adaptor and display (24-bit recommended)

Sun Solaris Run-Time Libraries. MATLAB dynamically links against the Solaris C++ Run-Time Library, which must reside on the same machine.

To determine whether this library is already installed, type the following line in a UNIX shell. (If you get a `No match` response, then you need to install it from your Solaris system CD.)

```
ls -l /usr/lib/libCstd.so.*
```

Accessibility Notes

MathWorks products includes a number of modifications to make them more accessible to all users.

For installation instructions relating to accessibility support, see “Installation Notes for Accessibility Support” on page 42.

Products Updated

The MathWorks has made general modifications to make its products more accessible. Particular emphasis has been placed on the accessibility of the following products/features:

- The product installation process
- MATLAB®
- Simulink®
- Control System Toolbox
- Curve Fitting Toolbox
- Excel® Link
- MATLAB® Compiler
- Optimization Toolbox
- Signal Processing Toolbox
- Statistics Toolbox

Summary of Accessibility Support

Accessibility support for blind and visually impaired users includes

- Support for screen readers and screen magnifiers, as described in “Assistive Technologies” on page 38
- Command-line alternatives for most graphical user interface (GUI) options
- Keyboard access to GUI components
- A clear indication of the current cursor focus
- Information available to assistive technologies about user interface elements, including the identity, operation, and state of the element

- Nonreliance on color coding as the sole means of conveying information about working with a GUI
- Noninterference with user-selected contrast and color selections and other individual display attributes, as well as noninterference for other operating system-level accessibility features
- Consistent meaning for bit-mapped images used in GUIs
- HTML documentation that is accessible to screen readers

The MathWorks believes that its products do not rely on auditory cues as the sole means of conveying information about working with a GUI. However, if you do encounter any issues in this regard, please report them to the MathWorks Technical Support group.

```
http://www.mathworks.com/contact_TS.html
```

Keyboard access to the user interface includes support for "sticky keys," which allow you to press key combinations (such as **Ctrl+C**) sequentially rather than simultaneously.

Except for scopes and real-time data acquisition, the MathWorks software does not use flashing or blinking text, objects, or other elements having a flash or blink frequency greater than 2 Hz and lower than 55 Hz.

Accessibility Support Details

These notes about product accessibility cover the following topics:

- "Assistive Technologies" on page 38
- "Troubleshooting" on page 38
- "Documentation" on page 40

If you are using a screen reader, such as JAWS, see also "Installation Notes for Accessibility Support" on page 42.

Assistive Technologies

Note To take advantage of accessibility support features, you must use MathWorks products on a Microsoft Windows platform.

Tested Assistive Technologies

For R2006a, The MathWorks has tested the following assistive technologies:

- JAWS 4.5 and 5.0 (recommended) for Windows (screen reader) from Freedom Scientific
- Built-in accessibility aids from Microsoft, including the Magnifier and "sticky keys"

Use of Other Assistive Technologies

Although The MathWorks has not tested other assistive technologies, such as other screen readers or ZoomText Xtra (screen magnifier) from Ai Squared, The MathWorks believes that most of the accessibility support built into its products should work with most assistive technologies that are generally similar to the ones tested.

If you use other assistive technologies than the ones tested, The MathWorks is very interested in hearing from you about your experiences.

Troubleshooting

This section identifies workarounds for some possible issues you may encounter related to accessibility support in MathWorks products.

JAWS Does Not Detect When the MATLAB Installation Has Started

When you select `setup.exe`, the Windows copying dialog box opens and you are informed. After the files have been copied, the installation splash screen opens, and then the installer starts. However, JAWS does not inform you that the installer has begun: the installer either starts up below other windows or applications or it is minimized. Since the installer is not an active item, nothing is read.

Therefore, check the Windows applications bar for the installer. After you go to the installer, you can use JAWS to perform the installation.

JAWS Stops Speaking

When many desktop components are open, JAWS with MATLAB sometimes stops speaking.

If this happens, close most of the desktop components, exit MATLAB, and restart.

Command Output Not Read

In the MATLAB Command Window, JAWS does not automatically read the results of commands.

To read command output, first select **File > Preferences > Command Window**, select the option **Use arrow keys for navigation instead of command history recall**, and click **OK**. Then, in the Command Window, press the arrow keys to move to the command output and use JAWS keystrokes to read the output.

With this preference set, you cannot use arrow keys to recall previous commands. Instead use the following key bindings:

- Windows key bindings:
  - Previous history: **Ctrl+up arrow**
  - Next history: **Ctrl+down arrow**
- Emacs key bindings:
  - Previous history: **Ctrl+p**
  - Next history: **Ctrl+n**

To return to using the up and down arrow keys to recall previous commands, clear the preference.

Some GUI Menus Are Treated as Check Boxes

For some GUIs (for example, the figure window), menus are treated by JAWS as though they are check boxes, whether or not they actually are.

You can choose a menu item for such GUIs by using accelerator keys (e.g., **Ctrl+N** to select **New Figure**), if one is associated with a menu item. You can also use mnemonics for menu navigation (e.g., **Alt+E**).

Note that check boxes that you encounter by tabbing through the elements of a GUI are handled properly.

Text Ignored in Some GUIs

For some dialog boxes, JAWS reads the dialog box title and any buttons, but ignores any text in the dialog box.

Also, in parts of some GUIs, such as some text-entry fields, JAWS ignores the label of the field. However, JAWS will read any text in the text box.

Documentation

Documentation is available in HTML format for all MathWorks products that are included in R2006a.

Accessing the Documentation

To access the documentation with a screen reader, go to the R2006a documentation area on the MathWorks Web site at

```
http://www.mathworks.com/access/helpdesk/help/helpdesk.html
```

Navigating the Documentation

Note that the first page that opens lists the products. To get the documentation for a specific product, click the link for that product.

The table of contents is in a separate frame. You can use a document's table of contents to navigate through the sections of that document.

Because you will be using a general Web browser, you will not be able to use the search feature included in the MATLAB Help browser. You will have access to an index for the specific document you are using. The cross-product index of the MATLAB Help browser is not available when you are using a general Web browser.

Products

The documentation for all products is in HTML and can be read with a screen reader. However, for most products, most equations and most graphics are not accessible.

The following product documentation has been modified (as described below) to enhance its accessibility for people using a screen reader such as JAWS:

- MATLAB (many sections, but not the function reference pages (however, M-file help is accessible))
- Excel Link
- Optimization Toolbox
- Signal Processing Toolbox
- Statistics Toolbox

Documentation Modifications

Modifications to the documentation include the following:

- Describing illustrations in text (either directly or via links)
- Providing text to describe the content of tables (as necessary)
- Restructuring information in tables to be easily understood when a screen reader is used
- Providing text links in addition to any image mapped links

Equations

Equations that are integrated in paragraphs are generally explained in words. However, most complex equations that are represented as graphics are not currently explained with alternative text.

Installation Notes for Accessibility Support

Note If you are not using a screen reader such as JAWS, you can skip this section.

This section describes the installation process for setting up your MATLAB environment to work effectively with JAWS.

Use the regular MATLAB installation script to install the products for which you are licensed. The installation script has been modified to improve its accessibility for all users.

Note Java Access Bridge 1.1 is installed automatically when you install MATLAB.

After you complete the product installation, there are some additional steps you need to perform to ensure JAWS works effectively with MathWorks products.

Setting Up JAWS

Make sure that JAWS is installed on your machine. If it is, there is probably a shortcut to it on the Windows desktop.

Setting up JAWS involves these tasks:

1. Add the Access Bridge to your Windows path (for networked installations only).
2. Create the `accessibility.properties` file.

These tasks are described in more detail below.

(For Networked Installations Only) Add Access Bridge to Your Path. If you are running MATLAB in a networked installation environment (that is, if the MATLAB Installer was not run on your machine), you need to take the following steps to add Access Bridge to your Windows path.

Note This procedure assumes your Windows **Start** button is set to Classic mode. To set Classic mode, from the **Start** button, select **Settings**. Next select **Task Bar and Menu**. Then select the **Start Menu** tab and make sure the `Classic Start Menu` option is enabled. Click **OK** and you are done.

1. From the **Start** button, select **Settings**, next select **Control Panel**. Scroll down and click the **System** icon to display the System Properties dialog box.
2. In the System Properties dialog box, select the **Advanced** tab.
3. Click **Environment Variables**.
4. Under **System variables**, select the `Path` option.
5. Click the **Edit** button.
6. To the start of the `Path` environment variable, add the directory that contains `matlab.exe`; for example:

   ```
   C:\matlab71\bin\win32;
   ```

   Be sure to include that semicolon between the end of this directory name and the text that was already there.
7. Click **OK** three times.
8. If JAWS is already running, exit and restart.

Note JAWS must be started with these path changes in effect to work properly with MATLAB.

Create the accessibility.properties File.

1. Create a text file that contains the following two lines:

   ```
   screen_magnifier_present=true
   assistive_technologies=com.sun.java.accessibility.AccessBridge
   ```

2 Use the filename `accessibility.properties`.

3 Move the `accessibility.properties` file into

```
$(matlabroot)\sys\java\jre\win32\jre1.5.0\lib\
```

JAWS Pronunciation Dictionary. As a convenience, The MathWorks provides a pronunciation dictionary for JAWS. This dictionary is in a file called `MATLAB.jdf`.

During installation, the file is copied to your system under the MATLAB root directory at `sys\Jaws\matlab.jdf`.

To use the dictionary, you must copy it to the `\SETTINGS\ENU` folder located beneath the JAWS root installation directory.

You need to restart JAWS and MATLAB for the settings to take effect.

Testing

After you install JAWS and set up your environment as described above, you should test to ensure JAWS is working properly:

1 Start JAWS.

2 Start MATLAB.

JAWS should start talking to you as you select menu items and work with the MATLAB user interface in other ways.

Quick Access to Product-Specific Release Notes

This section provides online links to the product-specific release notes for each MathWorks product. If a product is *not* listed below, it has not changed significantly since Release 11.

MATLAB and Simulink are listed first in the table. The rest of the MathWorks products are listed alphabetically.

Note Products followed by an asterisk (*) are updated since R14SP3. Refer to release notes for products that do *not* have an asterisk only if you are upgrading from a release before R14SP3.

- MATLAB®*
- Simulink®*
- Aerospace Blockset*
- Bioinformatics Toolbox*
- CDMA Reference Blockset
- Communications Blockset*
- Communications Toolbox*
- Control System Toolbox*
- Curve Fitting Toolbox
- Data Acquisition Toolbox*
- Database Toolbox
- Datafeed Toolbox*
- Distributed Computing Toolbox*
- Embedded Target for Infineon® C166 Microcontrollers
- Embedded Target for Motorola® HC12
- Embedded Target for Motorola® MPC555*
- Embedded Target for OSEK/VDX®
- Embedded Target for TI C2000™ DSP*
- Embedded Target for TI C6000™ DSP*

- Excel Link
- Filter Design HDL Coder*
- Filter Design Toolbox*
- Financial Derivatives Toolbox
- Financial Time Series Toolbox (now part of the Financial Toolbox)
- Financial Toolbox*
- Fixed-Income Toolbox
- Fixed-Point Toolbox*
- Fuzzy Logic Toolbox (no release notes)
- GARCH Toolbox*
- Gauges Blockset*
- Genetic Algorithm and Direct Search Toolbox
- Image Acquisition Toolbox*
- Image Processing Toolbox*
- Instrument Control Toolbox*
- Link for Code Composer Studio™ Development Tools*
- Link for ModelSim®*
- Mapping Toolbox*
- MATLAB® Builder for COM (now part of MATLAB® Builder for .NET)
- MATLAB® Builder for Excel
- MATLAB® Builder for .NET (new product)
- MATLAB® Compiler*
- MATLAB® Distributed Computing Engine*
- MATLAB® Report Generator
- MATLAB® Web Server
- Model-Based Calibration Toolbox*
- Model Predictive Control Toolbox
- Neural Network Toolbox*
- OPC Toolbox*
- Optimization Toolbox*
- Partial Differential Equation Toolbox* (no release notes)

- Real-Time Windows Target
- Real-Time Workshop®*
- Real-Time Workshop® Embedded Coder*
- RF Blockset*
- RF Toolbox*
- Robust Control Toolbox*
- Signal Processing Blockset*
- Signal Processing Toolbox*
- SimBiology (new product)
- SimDriveline*
- SimEvents* (new product)
- SimMechanics*
- SimPowerSystems*
- Simulink® Accelerator*
- Simulink® Control Design*
- Simulink® Fixed Point*
- Simulink® Parameter Estimation
- Simulink® Report Generator*
- Simulink® Response Optimization*
- Simulink® Verification and Validation*
- Spline Toolbox*
- Stateflow® and Stateflow® Coder*
- Statistics Toolbox*
- Symbolic Math Toolbox
- System Identification Toolbox*
- Video and Image Processing Blockset*
- Virtual Reality Toolbox*
- Wavelet Toolbox
- xPC Target*
- xPC TargetBox®

MATLAB Software Acknowledgments

MATLAB and/or its associated products include software developed by the following third parties.

ARnoldi PACKage (ARPACK)

Rich Lehoucq, Kristi Maschhoff, Danny Sorensen, and Chao Yang
`http://www.caam.rice.edu/software/ARPACK`

Assertion blocks were developed in cooperation with

Helmut Keller, Andreas Rau, and Joachim Boensch, members of the Control System Design (CSD) group at DaimlerChrysler Germany.

Automatically Tuned Linear Algebra Software (ATLAS)

R. Clint Whaley and Jack Dongarra
`http://www.netlib.org/atlas`

The Image Acquisition Toolbox DCAM adaptor uses the Carnegie Mellon University driver to communicate with cameras compatible with the IIDC 1394-based Digital Camera Specification (DCAM).

CopyrightC 2003 Carnegie Mellon University All Rights Reserved
The Robotics Institute - Carnegie Mellon University
`http://www-2.cs.cmu.edu/~iwan/1394/`

FDLIBM C math library for machines that support IEEE 754 floating point

Developed at SunSoft, a Sun Microsystems, Inc. business, by Kwok C. Ng and others. FDLIBM is freely redistributable and is available through NetLib. For information about FDLIBM, see `http://www.netlib.org`.

`fft` and related MATLAB functions are based on the FFTW library.

Developed by Matteo Frigo and Steven G. Johnson
Copyright ” 2003 Matteo Frigo
Copyright ” 2003 Massachusetts Institute of Technology. All rights reserved.
Used under terms of a commercial license
`http://www.fftw.org`

FreeType2 Project library is included with Simulink.

FreeType was created by David Turner, Robert Wilhelm, and Werner Lemberg http://freetype.org

A few MathWorks products contain the `graphviz` code from AT&T. ("AT&T Software") proprietary to AT&T Corp. ("AT&T").

The AT&T Software is provided to you "AS IS". YOU ASSUME TOTAL RESPONSIBILITY AND RISK FOR USE OF THE AT&T SOFTWARE. AT&T DOES NOT MAKE, AND EXPRESSLY DISCLAIMS, ANY EXPRESS OR IMPLIED WARRANTIES OF ANY KIND WHATSOEVER, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, WARRANTIES OF TITLE OR NON-INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHTS, ANY WARRANTIES ARISING BY USAGE OF TRADE, COURSE OF DEALING OR COURSE OF PERFORMANCE, OR ANY WARRANTY THAT THE AT&T SOFTWARE IS "ERROR FREE" OR WILL MEET YOUR REQUIREMENTS

Unless you accept a license to use the AT&T Software, you shall not reverse compile, disassemble or otherwise reverse engineer this product to ascertain the source code for any AT&T Software.

HDF capability in the functions `imread`, `imwrite`, `imfinfo`, and `hdf` and HDF 5 capability in the functions `hdf5info` and `hdf5read` are based on code of which portions were developed at

> The National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign.

J2PrinterWorks `.class` files are a product of Wildcrest Associates.

JPEG capability in the functions `imread`, `imwrite`, `imfinfo`, `print`, and `saveas`:

> This software is based in part on the work of the Independent JPEG Group.

Linear Algebra PACKage (LAPACK)

> `http://www.netlib.org/lapack` (for general information about LAPACK)
> For details, see the *LAPACK User's Guide*.
> E. Anderson, Z. Bai, C. Bischof, L. S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen
> For a printed version of the *LAPACK User's Guide*, go to `http://www.siam.org`.
> For an online version of the *LAPACK User's Guide,* go to `http://www.netlib.org/lapack/lug/lapack_lug.html`.
>
> Acknowledgements: This material is based upon work supported by the National Science Foundation under Grant No. ASC-9313958 and DOE Grant No. DE-FG03-94ER25219. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation (NSF) or the Department of Energy (DOE). We give credit to and thank all of the LAPACK authors for allowing the reuse of large portions of the LAPACK Users' Guide.

openVRML, developed by The OpenVRML project (www.openvrml.org), is used in the Virtual Reality Toolbox. openVRML is redistributed herein under The GNU Lesser General Public License (LGPL), Version 2.1.

Qhull based computational geometry capability in MATLAB

>

Sparse matrix minimum degree permutation functions `colamd` and `symamd`

>
> Authors of the code are Stefan I. Larimore and Timothy A. Davis (`davis@cise.ufl.edu`), University of Florida. The algorithm was developed in collaboration with John Gilbert, Xerox PARC, and Esmond Ng, Oak Ridge National Laboratory.
> This work was supported by the National Science Foundation, under grants DMS-9504974 and DMS-9803599.
> For complete copyright information, issue the MATLAB command `edit colamd` or `edit symamd`.

The SLICOT library of numerical algorithms for computations in systems and control theory is used in the Control System Toolbox. The SLICOT library is developed by the NICONET group (`www.win.tue.nl/niconet/NIC2/slicot.html`).

> More detailed information on SLICOT can be found in:
> Benner, P., Mehrmann, V., Sima, V., Van Huffel, S., and A. Varga: "SLICOT - A Subroutine Library in Systems and Control Theory", June 1997, NICONET Report 97-3.
> SLICOT is freely available through WWW: (http://www.win.tue.nl/wgs/) or anonymous ftp: (`ftp://wgs.esat.kuleuven.ac.be/pub/WGS/SLICOT/`).

The MATLAB implementation of TeX is compiled from Donald Knuth's original TeX parser (Version: 3.14159) located on the TeX Archive Network: `www.ctan.org`. The LaTeX distribution was also obtained from `www.ctan.org`.

TIFF capability in the functions `imread`, `imwrite`, `imfinfo`, `print`, and `saveas`:

>

Unsymmetric MultiFrontal PACKage (UMFPACK) for solving unsymmetric sparse linear systems.

See `http://www.cise.ufl.edu/research/sparse/umfpack` for general information about UMFPACK. For details, the *UMFPACK Version 4.6 User Guide* is available at `http://www.cise.ufl.edu/research/sparse/umfpack/v4.6/UMFPACK/Doc/UserGuide.pdf`.

wxWidgets cross-platform native UI framework, developed by the wxWidgets project (www.wxwidgets.org), is used in the Virtual Reality Toolbox. wxWidgets is redistributed herein under The wxWidgets Library License, Version 3.0. The wxWidgets Libarary License does not apply to any other part of MATLAB, Simulink, or other products or services distributed by the The MathWorks, Inc.

MATLAB includes an XML parser developed by the Apache Software Foundation (`www.apache.org`) redistributed herein under The Apache Software License, Version 1.1. The Apache Software License does not apply to any other part of MATLAB, Simulink, or other products or services distributed by The MathWorks, Inc.

Revision: 1.1.2.1
Copyright Notice and Statement for NCSA Hierarchical Data Format (HDF)
Software Library and Utilities

Contributors: National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign (UIUC), Lawrence Livermore National Laboratory (LLNL), Sandia National Laboratories (SNL), Los Alamos National Laboratory (LANL), Jean-loup Gailly and Mark Adler (gzip library).

or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.

Portions of HDF5 were developed with support from the University of California, Lawrence Livermore National Laboratory (UC LLNL).
The following statement applies to those portions of the product and must be retained in any redistribution of source code, binaries, documentation, and/or accompanying materials:

This work was partially produced at the University of California, Lawrence Livermore National Laboratory (UC LLNL) under contract no. W-7405-ENG-48 (Contract 48) between the U.S. Department of Energy (DOE) and The Regents of the University of California (University) for the operation of UC LLNL.

DISCLAIMER:

This work was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately-owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2003 International Business Machines Corporation and others All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

The Distributed Computing Toolbox and MATLAB Distributed Computing Engine include software developed by the following third parties.

The Java Service Wrapper by Tanuki Software.

Copyright (c) 1999, 2004 Tanuki Software

Permission is hereby granted, free of charge, to any person obtaining a copy of the Java Service Wrapper and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sub-license, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.