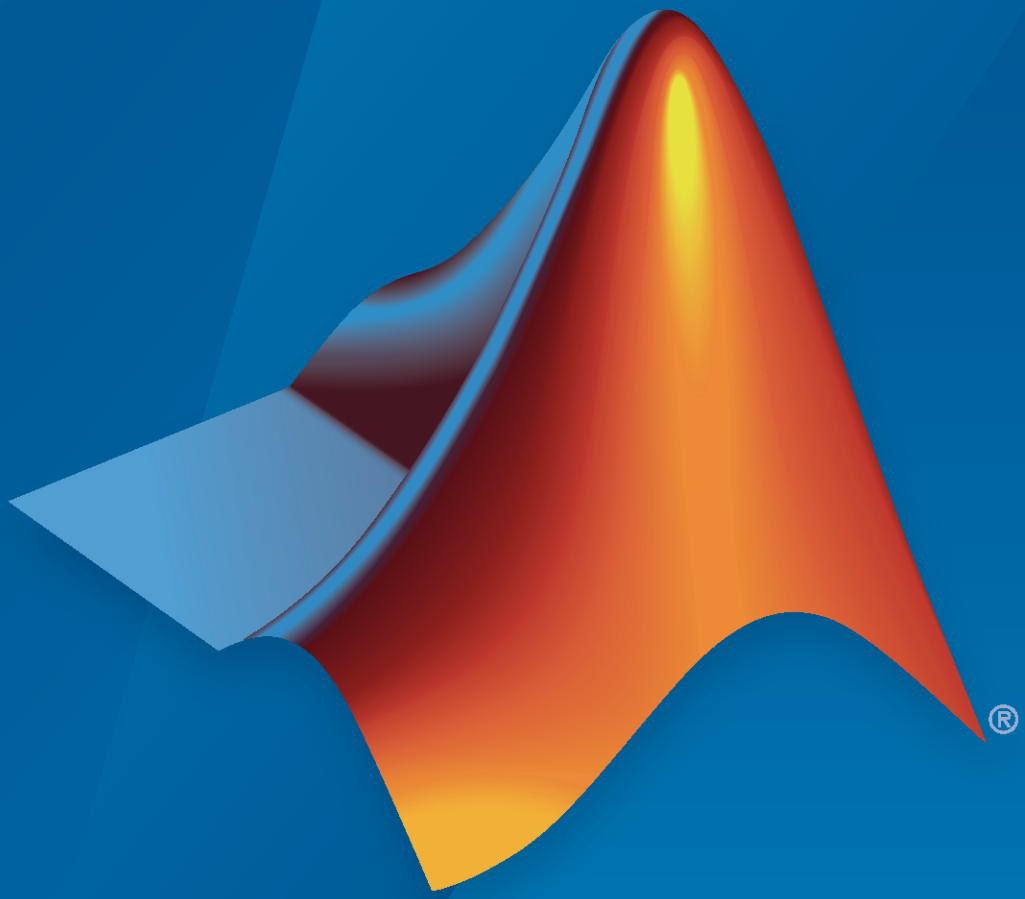
MATLAB® Compiler™ Release Notes



MATLAB®

MathWorks®

How to Contact MathWorks

Latest news: `www.mathworks.com`

Sales and services: `www.mathworks.com/sales_and_services`

User community: `www.mathworks.com/matlabcentral`

Technical support: `www.mathworks.com/support/contact_us`

Phone: 508-647-7000

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

MATLAB® Compiler™ Release Notes

Contents

R2025b

R2025a

R2024b

R2024a

R2023b

R2023a

R2022b

R2022a

R2021b

R2021a

R2025b

Version: 25.2

Bug Fixes

Quality and stability improvements

R2025b delivers quality and stability improvements, building on the new features introduced in R2025a.

R2025a

Packaging: Package code using new Compiler apps

The Compiler apps have been updated to integrate with MATLAB projects. Compiler apps let you customize the packaging process using a graphical approach, deploy to multiple deployment targets with compiler tasks, and export your packaging settings as a deployment script.

These Compiler apps are available:

- Standalone Application Compiler — Create standalone applications and standalone Windows® applications.
- Excel Add-in Compiler — Create Excel® add-ins.
- Web App Compiler — Create web apps for MATLAB Web App Server™.
- C Shared Library Compiler (MATLAB Compiler SDK) — Create C shared libraries.
- C++ Shared Library Compiler (MATLAB Compiler SDK) — Create C++ shared libraries.
- .NET Assembly Compiler (MATLAB Compiler SDK) — Create .NET assemblies.
- COM Component Compiler (MATLAB Compiler SDK) — Create COM components.
- Java Package Compiler (MATLAB Compiler SDK) — Create Java® packages.
- Python Package Compiler (MATLAB Compiler SDK) — Create Python® packages.
- Production Server Archive Compiler (MATLAB Compiler SDK) — Create archives for MATLAB Production Server™.

For more information, see Choose Deployment Option.

Compatibility Considerations

In R2024b and earlier releases, the compiler apps place generated output files in the folders `for redistribution`, `for redistribution_files_only`, and `for testing`. Starting in R2025a, the compiler apps create a folder named after the compiler task in the project root folder and place generated files in a subfolder named `output`.

MATLAB Runtime: Exclude optional run-time dependencies in installer

You can now use the `compiler.package.installer`, `compiler.runtime.customInstaller`, `compiler.runtime.createDockerImage`, and `compiler.package.microserviceDockerImage` (MATLAB Compiler SDK) functions with the `OptionalDependencies` option to control whether the installer includes optional run-time dependencies. This option allows you to minimize the size of the generated installer.

compiler.build.Results: New property to identify MATLAB Runtime dependencies

The new `RuntimeDependencies` property for `compiler.build.Results` identifies optional and required dependencies for your build.

⚠Functionality Being Removed or Changed

deploytool function has been removed

The `deploytool` function has been removed. To open a Compiler app, select the app from the **Apps** menu or use a function such as `standaloneApplicationCompiler` or `webAppCompiler`. For more information, see Choose Deployment Option.

Compiler apps have new output file folder structure

Behavior change

The Compiler apps no longer generate output files in the folders `for redistribution`, `for redistribution_files_only`, and `for testing`. By default, the compiler apps create a folder named after the compiler task in the project root folder and place generated files in a subfolder named `output`.

Hadoop Compiler app and hadoopCompiler function have been removed

The Hadoop Compiler app and the `hadoopCompiler` function have been removed. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function have been removed

The `-build` and `-package` options in the `applicationCompiler` function have been removed in a future release. To build applications, use the `compiler.build.standaloneApplication` function or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `compiler.package.installer` function will not be changed.

-package option in the webAppCompiler function has been removed

Warns

The `-package` option in the `webAppCompiler` function has been removed. To package MATLAB apps into web apps, use the `compiler.build.webAppArchive` function or the Web App Compiler app.

MATLAB Runtime support for Apache Ambari has been removed

The ability to download MATLAB Runtime from within Apache® Ambari™ has been removed.

MATLAB Compiler support has been removed from deep learning training functions and function for estimating lidar to camera transformation

Errors

Support for using MATLAB Compiler has been removed for these functions:

- `trainVoxelRCNNObjectDetector`
- `trainPointPillarsObjectDetector`
- `trainRandlanet`
- `estimateLidarCameraTransform`

R2024b

Version: 24.2

New Features

Bug Fixes

MATLAB Runtime: Create custom MATLAB Runtime installer

You can use the `compiler.runtime.customInstaller` function to create a MATLAB Runtime installer with a minimal size footprint that installs only the MATLAB Runtime components required to run the specified artifacts created with MATLAB Compiler. The custom installer includes the minimum set of MATLAB Runtime components necessary to run one or more applications created with MATLAB Compiler or MATLAB Compiler SDK™.

Packaging: Specify encryption key for packaging MATLAB code using compiler.build functions

You can use any `compiler.build` function (except for `compiler.build.excelClientForProductionServer`) with the `ExternalEncryptionKey` option to specify a 256-bit AES encryption key and a MEX-file loader interface to retrieve the decryption key at runtime. This option is equivalent to the `mcc -k` option.

★Security: Package code with secrets using compiler.build functions

You can use any `compiler.build` function (except for `compiler.build.excelClientForProductionServer`) with the `SecretsManifest` option to include a JSON file that specifies secrets to embed within your deployable code archive. This option is equivalent to the `mcc -J` option.

For more information on deployment using secrets, see Handle Sensitive Information in Deployed Applications.

MATLAB Runtime: Optional product dependency on GPU runtime libraries

GPU libraries are no longer required dependencies for MATLAB Runtime. Packaged MATLAB code that does not display graphical output can be run using a MATLAB Runtime installation without GPU libraries.

If you use the `compiler.runtime.customInstaller` function to create a MATLAB Runtime installer for MATLAB code without any graphical dependencies, the resulting installer will not include GPU libraries.

A MATLAB Runtime Docker® image without GPU support is available in the MathWorks® Docker repository. For details, see MATLAB Runtime Container.

MATLAB Runtime: Component cache size increase and new location

When a compiled application runs, the files in the deployable archive (CTF file) are extracted to the MATLAB Runtime component cache. The default maximum cache size has increased from 32 MB to 1024 MB. In addition, the default component cache location has changed according to the following table:

Platform	Old Cache Location	New Cache Location
Windows	`%LOCALAPPDATA%\Temp\ %USERNAME% \mcrCache<version>`	`%LOCALAPPDATA%\MathWorks \MatlabRuntimeCache \<release_version>`
Linux®	`$HOME/.mcrCache<version>`	`$HOME/.MathWorks/ MatlabRuntimeCache/ <release_version>`
Mac	A cache is not used in MATLAB R2020a and later. The app bundle contains the files necessary for run time.	

For more information about the component cache, see MATLAB Runtime Component Cache and Deployable Archive Embedding.

R2024a

★Security: Package code with secrets

You can avoid exposing sensitive information, such as passwords, in code by using secrets and a local MATLAB vault. Your MATLAB vault provides encrypted and persistent storage for secrets, and it is accessible only from the exact combination of your operating system account and local machine.

When packaging MATLAB code, you can use the functionality provided by the MATLAB vault in deployed applications and web apps. To retrieve secret keys from your MATLAB vault, use the `getSecret` function in deployed code.

Package secrets with your application by using the `mcc -J` option to include a JSON file that specifies secrets to embed within your deployable code archive. For more information on deployment using secrets, see Handle Sensitive Information in Deployed Applications.

For an example on deploying a standalone application with secret keys, see Access Sensitive Information in Standalone Application.

Application Installers: Create application installer without MATLAB Runtime

You can use the `compiler.package.installer` function with the `RuntimeDelivery` option set to `'none'` to create an application installer that does not include MATLAB Runtime. Use this option if your end user will install MATLAB Runtime using another method.

Application Installers: Create ZIP installer

On Windows, you can use the `compiler.package.installer` function with the `PackageType` option to specify whether to force creation of a ZIP file that contains the installer. By default, a ZIP installer is created only when the installer size exceeds 2 GB.

Packaging: Include MATLAB packages in deployed application

You can package and deploy MATLAB code that uses MATLAB packages and MATLAB Toolbox packages.

Excel Integration: Support for MinGW 8.1 and Windows SDK for Windows 11

MATLAB Compiler now supports MinGW® 8.1 and the Windows SDK for Windows 11 for creating Excel add-ins. For a complete list of supported compilers for MATLAB Compiler, see `https://www.mathworks.com/support/requirements/supported-compilers.html`.

⚠Functionality Being Removed or Changed

Hadoop Compiler app and hadoopCompiler function will be removed

Warns

The Hadoop Compiler app and the `hadoopCompiler` function will be removed in a future release. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function will be removed
Warns

The `-build` and `-package` options in the `applicationCompiler` function will be removed in a future release. To build applications, use the `compiler.build.standaloneApplication` function or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `compiler.package.installer` function will not be changed.

The Application Compiler app will also not be changed. You can continue to the use the app to build applications.

-package option in the webAppCompiler function will be removed
Warns

The `-package` option in the `webAppCompiler` function will be removed in a future release. To package MATLAB apps into web apps, use the `compiler.build.webAppArchive` function or the Web App Compiler app.

-build and -package options in the deploytool function will be removed
Warns

The `-build` and `-package` options in the `deploytool` function will be removed in a future release. To build applications, use one of the `compiler.build` family of functions or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `deploytool` function will not be changed.

MATLAB Runtime support for Apache Ambari will be removed
Warns

The ability to download MATLAB Runtime from within Apache Ambari will be removed in a future release.

R2023b

Web Apps and Standalone Applications: Data brushing supported in graphics

Graphics created in web apps and standalone applications support data brushing. Use data brushing in these applications just as you would in MATLAB figures.

Docker Integration: Create MATLAB Runtime Docker image

On Linux, you can use the `compiler.runtime.createDockerImage` function to create a custom MATLAB Runtime image that can run multiple applications. To use the image with a Docker container, specify the MATLAB Runtime image name as an input to the `compiler.package.docker` or `compiler.package.microserviceDockerImage` (MATLAB Compiler SDK) function.

⚠Functionality Being Removed or Changed

Hadoop Compiler app and hadoopCompiler function will be removed

Warns

The Hadoop Compiler app and the `hadoopCompiler` function will be removed in a future release. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function will be removed

Warns

The `-build` and `-package` options in the `applicationCompiler` function will be removed in a future release. To build applications, use the `compiler.build.standaloneApplication` function or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `compiler.package.installer` function will not be changed.

The Application Compiler app will also not be changed. You can continue to the use the app to build applications.

-package option in the webAppCompiler function will be removed

Warns

The `-package` option in the `webAppCompiler` function will be removed in a future release. To package MATLAB apps into web apps, use the `compiler.build.webAppArchive` function or the Web App Compiler app.

-build and -package options in the deploytool function will be removed

Warns

The `-build` and `-package` options in the `deploytool` function will be removed in a future release. To build applications, use one of the `compiler.build` family of functions or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `deploytool` function will not be changed.

MATLAB Runtime support for Apache Ambari will be removed

Warns

The ability to download MATLAB Runtime from within Apache Ambari will be removed in a future release.

R2023a

Version: 8.6

Bug Fixes

Compatibility Considerations

Functionality Being Removed or Changed

MATLAB Compiler Application Installers: -tmpdir removed

The option to specify where temporary files are stored during installation using the `-tmpdir` option has been removed.

Hadoop Compiler app and hadoopCompiler function will be removed

Warns

The Hadoop Compiler app and the `hadoopCompiler` function will be removed in a future release. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function will be removed

Warns

The `-build` and `-package` options in the `applicationCompiler` function will be removed in a future release. To build applications, use the `compiler.build.standaloneApplication` function or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `compiler.package.installer` function will not be changed.

The Application Compiler app will also not be changed. You can continue to the use the app to build applications.

-package option in the webAppCompiler function will be removed

Warns

The `-package` option in the `webAppCompiler` function will be removed in a future release. To package MATLAB apps into web apps, use the `compiler.build.webAppArchive` function or the Web App Compiler app.

-build and -package options in the deploytool function will be removed

Warns

The `-build` and `-package` options in the `deploytool` function will be removed in a future release. To build applications, use one of the `compiler.build` family of functions or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `deploytool` function will not be changed.

MATLAB Runtime support for Apache Ambari will be removed

Warns

The ability to download MATLAB Runtime from within Apache Ambari will be removed in a future release.

R2022b

MATLAB Runtime: Installation path updated to use MATLAB release name

MATLAB Runtime now installs to a folder whose name uses the corresponding MATLAB release name.

Previously, the default path to an installation of MATLAB Runtime included the MATLAB Runtime version, such as `v912` in R2022a. The new installer uses the MATLAB release name in the installation path, for example, `C:\Program Files\MATLAB\MATLAB Runtime\R2022b`.

Application Deployment: Create a Deployment Script using the createDeploymentScript function

You can create a MATLAB deployment script from a MATLAB Compiler project file (`.prj`) using the `createDeploymentScript` function. This function converts the project deployment settings to the corresponding `compiler.build` commands. The generated MATLAB script creates a compiled component using the same settings as the MATLAB Compiler project.

Packaging: Obfuscate all MATLAB code prior to packaging

You can use the `mcc` command with the `-j` option to automatically convert all `.m` files to P-files before packaging. This option generates a P-code file with a `.p` extension for each `.m` file included in the `mcc` command. P-code files are an obfuscated, execute-only form of MATLAB code. For more details, see `pcode`.

Packaging: Specify encryption key for packaging MATLAB code

You can use the `mcc` command with the `-k` option to specify an AES encryption key and a MEX-file loader interface to retrieve the decryption key at runtime. If you do not specify any arguments after `-k`, `mcc` generates a 256-bit AES key and a loader MEX-file.

⚠Apache Spark 3.x Support

When you create an Apache Spark™ application using `mcc`, the compiler uses Spark 3.x by default. You can specify other versions of Spark in the `mcc` command.

⚠Functionality Being Removed or Changed

MATLAB Runtime Installer: Automated Mode removed

The option to install MATLAB Runtime noninteractively using the `-mode automated` option has been removed.

You can install MATLAB Runtime in silent mode by using installer command-line options or an installer control file. In silent mode, the installer runs as a background task and does not display any dialog boxes.

MATLAB Compiler Application Installers: Automated Mode removed

The option to install packaged MATLAB Compiler applications noninteractively using the `-mode automated` option has been removed.

You can install applications in silent mode by using installer command-line options or an installer control file. In silent mode, the installer runs as a background task and does not display any dialog boxes.

Option to build and package MATLAB code from within the Function Wizard for Excel add-ins has been removed

The option to build and package MATLAB code from within the Function Wizard for Excel add-ins has been removed. To create an Excel add-in, use the `compiler.build.excelAddIn` function or the Library Compiler app.

Hadoop Compiler app and hadoopCompiler function will be removed
Warns

The Hadoop Compiler app and the `hadoopCompiler` function will be removed in a future release. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function will be removed
Warns

The `-build` and `-package` options in the `applicationCompiler` function will be removed in a future release. To build applications, use the `compiler.build.standaloneApplication` function or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `applicationCompiler` function will not be changed.

The Application Compiler app will also not be changed. You can continue to the use the app to build applications.

-package option in the webAppCompiler function will be removed
Warns

The `-package` option in the `webAppCompiler` function will be removed in a future release. To package MATLAB apps into web apps, use the `compiler.build.webAppArchive` function or the Web App Compiler app.

-build and -package options in the deploytool function will be removed
Warns

The `-build` and `-package` options in the `deploytool` function will be removed in a future release. To build applications, use one of the `compiler.build` family of functions or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. The other options of the `deploytool` function will not be changed.

MATLAB Runtime support for Apache Ambari will be removed
Warns

The ability to download MATLAB Runtime from within Apache Ambari will be removed in a future release.

R2022a

Web Apps: Customize web app behavior to each user

An app author using MATLAB Compiler to create a web app can customize the behavior of the app to each logged-in user. Customization involves any user-specific behavior the app author deems necessary. For example:

- Loading user-specific data or configurations to a web app
- Enabling user-specific UI components while disabling others

To retrieve user-specific information from a web app, you use the `compiler.UserInfo` function within the app. This function returns a `compiler.UserInfo` object containing at minimum the following properties: `UserID`, `DisplayName`, and `Groups`.

To retrieve more information, you configure additional properties in the `userinfo.json` file in the `webapps_private` folder on MATLAB Web App Server. The app author then uses the information from the `compiler.UserInfo` object to customize the web app to each user.

Customizing web app behavior to each user is supported in the MATLAB Web App Server product and not the development version of MATLAB Web App Server.

For details, see Customize Web App Behavior Based on User.

⚠mcc: Explicitly include MATLAB preferences when building and packaging MATLAB code for deployment

When using the `mcc` command to build and package MATLAB code for deployment, you must explicitly include MATLAB preferences using the `-a` option. For example, on a Windows system:

```
mcc -m helloWorld.m -a C:\Users\someuser\AppData\Roaming\MathWorks\MATLAB\R2025b\matlab.mlsetting
```

⚠Compatibility Considerations

In R2021b and earlier releases, when you used the `mcc` command to build and package MATLAB code for deployment, MATLAB preferences were automatically included.

⚠Functionality Being Removed or Changed

Option to build and package MATLAB code from within the Function Wizard for Excel add-ins has been removed

The option to build and package MATLAB code from within the Function Wizard for Excel add-ins has been removed. To create an Excel add-in, use the `compiler.build.excelAddIn` function or the Library Compiler app.

Hadoop Compiler app and hadoopCompiler function will be removed

Warns

The Hadoop Compiler app and the `hadoopCompiler` function will be removed in a future release. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function will be removed
Warns

The `-build` and `-package` options in the `applicationCompiler` function will be removed in a future release. To build applications, use the `compiler.build.standaloneApplication` function or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. There are no changes to the other options of the `applicationCompiler` function.

There are also no changes to the Application Compiler app. You can continue to the use the app to build applications.

-package option in the webAppCompiler function will be removed
Warns

The `-package` option in the `webAppCompiler` function will be removed in a future release. To package MATLAB apps into web apps, use the `compiler.build.webAppArchive` function or the Web App Compiler app.

-build and -package options in the deploytool function will be removed
Warns

The `-build` and `-package` options in the `deploytool` function will be removed in a future release. To build applications, use one of the `compiler.build` family of functions or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. There are no changes to the other options of the `deploytool` function.

MATLAB Runtime support for Apache Ambari will be removed
Warns

The ability to download MATLAB Runtime from within Apache Ambari will be removed in a future release.

R2021b

Version: 8.3

Bug Fixes

Compatibility Considerations

Packaging: Obfuscate file structures and file names

You can use the `mcc` command with the `-s` option to customize your archive for all deployment targets. This option obfuscates folder structures and file names in the deployable archive (`.ctf` file) from the end user.

Support Packages: Additional options for specifying support packages

You can use two new options to manually include support packages for all deployment targets.

- Use the `mcc` command with the `-Z` option to specify the method of adding support packages to the deployable archive.
- Use a `compiler.build` command with the `SupportPackages` option to specify the method of adding support packages.

To list installed support packages or those used by a specific file, see `compiler.codetools.deployableSupportPackages`.

⚠Spark Version Support for Spark Applications

Support	Spark Version	Platform
Added	Spark 3.0	Linux only

⚠Functionality Being Removed or Changed

Hadoop Compiler app and hadoopCompiler function will be removed
Warns

The Hadoop Compiler app and the `hadoopCompiler` function will be removed in a future release. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function will be removed
Warns

The `-build` and `-package` options in the `applicationCompiler` function will be removed in a future release. To build applications, use the `compiler.build.standaloneApplication` function or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. There are no changes to the other options of the `applicationCompiler` function.

There are also no changes to the Application Compiler app. You can continue to the use the app to build applications.

-package option in the webAppCompiler function will be removed
Warns

The `-package` option in the `webAppCompiler` function will be removed in a future release. To package MATLAB apps into web apps, use the `compiler.build.webAppArchive` function or the Web App Compiler app.

-build and -package options in the deploytool function will be removed
Warns

The `-build` and `-package` options in the `deploytool` function will be removed in a future release. To build applications, use one of the `compiler.build` family of functions or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. There are no changes to the other options of the `deploytool` function.

Option to build and package MATLAB code from within the Function Wizard for Excel add-ins will be removed
Warns

The option to build and package MATLAB code from within the Function Wizard for Excel add-ins will be removed in a future release. To create an Excel add-in, use the `compiler.build.excelAddIn` function or the Library Compiler app.

R2021a

Excel Integration: Create an Excel add-in package using the compiler.build.excelAddIn function

You can create a Microsoft® Excel add-in from the MATLAB command prompt using the `compiler.build.excelAddIn` function. This function, along with the `compiler.build.ExcelAddInOptions` function, provides an improved interface to specify various options associated with creating Excel add-ins. These functions are in addition to the existing Library Compiler app you use to create Excel add-ins.

Standalone Applications: Use thread-based parallel pool workers

You can create standalone applications that use thread-based parallel pool workers from Parallel Computing Toolbox™.

For more information, see Check Support for Thread-Based Environment (Parallel Computing Toolbox).

Web Apps: Use thread-based parallel pool workers

You can create web apps that use thread-based parallel pool workers from Parallel Computing Toolbox.

For more information, see Check Support for Thread-Based Environment (Parallel Computing Toolbox).

⚠Functionality Being Removed or Changed

Hadoop Compiler app and hadoopCompiler function will be removed

Warns

The Hadoop Compiler app and the `hadoopCompiler` function will be removed in a future release. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function will be removed

Warns

The `-build` and `-package` options in the `applicationCompiler` function will be removed in a future release. To build applications, use the `compiler.build.standaloneApplication` function or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. There are no changes to the other options of the `applicationCompiler` function.

There are also no changes to the Application Compiler app. You can continue to the use the app to build applications.

-package option in the webAppCompiler function will be removed

Warns

The `-package` option in the `webAppCompiler` function will be removed in a future release. To package MATLAB apps into web apps, use the `compiler.build.webAppArchive` function or the Web App Compiler app.

-build and -package options in the deploytool function will be removed

Warns

The `-build` and `-package` options in the `deploytool` function will be removed in a future release. To build applications, use one of the `compiler.build` family of functions or the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. There are no changes to the other options of the `deploytool` function.

Option to build and package MATLAB code from within the Function Wizard for Excel add-ins will be removed

Warns

The option to build and package MATLAB code from within the Function Wizard for Excel add-ins will be removed in a future release. To create an Excel add-in, use the `compiler.build.excelAddIn` function or the Library Compiler app.

R2020b

Standalone Applications: Create standalone applications using compiler.build.standaloneApplication and compiler.build.standaloneWindowsApplication functions

You can create standalone applications from the MATLAB command prompt using the new `compiler.build.standaloneApplication` and `compiler.build.standaloneWindowsApplication` functions. These functions, along with the `compiler.build.StandaloneApplicationOptions` function, provide an improved interface to specify various options associated with creating standalone applications. These functions complement the existing `mcc` command you use to create standalone applications.

The `compiler.build.standaloneApplication` function runs on any platform. The application extension is determined by the operating system you are running the function from.

The `compiler.build.standaloneWindowsApplication` function is available only on Windows operating systems.

Standalone Applications: Package standalone applications into Docker images on Linux operating systems

You can package standalone applications into Docker images using the `compiler.package.docker` function. To create a Docker image, you create a standalone application using the `compiler.build.standaloneApplication` function and pass the `Results` object generated by the function as an input argument to the `compiler.package.docker` function. You can modify options associated with creating a Docker image using the `compiler.package.DockerOptions` function. For details, see Package MATLAB Standalone Applications into Docker Images.

This functionality is supported only on Linux operating systems.

Web Apps: Create web apps using the compiler.build.webAppArchive function

You can create web app archives from the MATLAB command prompt using the `compiler.build.webAppArchive` function. This function, along with the `compiler.build.WebAppArchiveOptions` function, provides an improved interface to specify various options associated with creating web app archives. You can then copy or upload the web app archives to the server.

Simulink: Read and write data using Simulink.sdi.Run objects in deployed applications

You can use import functions for the Simulation Data Inspector in Simulink® to read and write data in applications deployed using MATLAB Compiler or Simulink Compiler. The import functions have built-in readers that import data from MAT, CSV, and Microsoft Excel files into `Simulink.sdi.Run` and `Simulink.sdi.Signal` objects. You can also use Simulation Data Inspector export functions to write the data to the application workspace, a MAT-file, or a Microsoft Excel file. These functions are supported for deployment:

- `Simulink.sdi.createRun`

- `Simulink.sdi.Run.create`
- `Simulink.sdi.addToRun`
- `add`
- `export` function for `Simulink.sdi.Signal` objects
- `export` function for `Simulink.sdi.Run` objects
- `Simulink.sdi.exportRun`

Functionality Being Removed or Changed

Hadoop Compiler app and hadoopCompiler function will be removed
Warns

The Hadoop Compiler app and the `hadoopCompiler` function will be removed in a future release. To create standalone MATLAB MapReduce applications or deployable archives from MATLAB map and reduce functions, use the `mcc` command.

-build and -package options in the applicationCompiler function will be removed
Warns

The `-build` and `-package` options in the `applicationCompiler` function will be removed in a future release. To build applications, use the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. There are no changes to the other options of the `applicationCompiler` function.

There are also no changes to the Application Compiler app. You can continue to the use the app to build applications.

-package option in the webAppCompiler function will be removed
Warns

The `-package` option in the `webAppCompiler` function will be removed in a future release. To package MATLAB apps into web apps, use the Web App Compiler app.

-build and -package options in the deploytool function will be removed
Warns

The `-build` and `-package` options in the `deploytool` function will be removed in a future release. To build applications, use the `mcc` command. To package and create an installer, use the `compiler.package.installer` function. There are no changes to the other options of the `deploytool` function.

Option to build and package MATLAB code from within the Function Wizard for Excel add-ins will be removed
Warns

The option to build and package MATLAB code from within the Function Wizard for Excel add-ins will be removed in a future release. To create an Excel add-in, use the Library Compiler app.