

Integrate MATLAB with External Programming Languages and Systems

MATLAB® provides a flexible, two-way integration with other programming languages, allowing you to reuse legacy code. For a list of programming languages and the supported versions, see [MATLAB Supported Interfaces to Other Languages](#).

Call C/C++ Code from MATLAB

MATLAB provides these features to help you integrate C/C++ algorithms into your MATLAB applications.

- A C/C++ shared library interface is a collection of functions dynamically loaded by an application at run time. Using a shared library has the advantage of packaging multiple library functions into one interface. In addition, MATLAB manages data type conversions.
 - Call C++ Library Functions - To call functions in a C++ shared library, use the `clib` package described in [C++ Libraries](#).
 - Whenever possible, choose the C++ interface over the C-only interface. For information about C++ support, see these [limitations](#).
 - To call functions in a C shared library, use the `calllib` function. For information, see [C Libraries](#). This feature works best with C-only libraries, but has these [limitations](#).

If you want more control over data conversion and memory management, consider writing a MEX file.

- A MEX file is wrapper code around a C/C++ algorithm that handles the conversion of MATLAB data types to C types. MEX files provide better performance than calling functions through MATLAB shared library interfaces. Also, MEX files give you more programmatic control over data conversion and memory management.
 - [C++ MEX Applications](#) use modern C++ programming features and, where possible, shared copies of data.
 - [C MEX File Applications](#) use the C matrix library and is supported for existing MEX functions. MathWorks recommends that whenever possible, choose C++ MEX over C MEX file applications.
 - If you have multiple functions in a library or do not have performance issues, consider writing a C++ library interface.

These features require C/C++ programming skills to create a library interface or to write a MEX function. However, you can give the resulting library or MEX function to any MATLAB user. The end user calls the functionality like any MATLAB function, without knowing the underlying details of the C/C++ language implementation.

To call MATLAB from a C/C++ language program, see [MATLAB Engine API for C++](#) or [MATLAB Engine API for C](#).

Use Objects from Other Programming Languages in MATLAB

If you have functions and objects in another programming language, you can call them from MATLAB. You do not need to be a software developer to integrate these objects into your MATLAB application. However, you need access to third-party documentation for the libraries.

MATLAB supports calling functions and using objects in the following languages.

- [C++ Libraries](#)
- [C Libraries](#)
- [MEX File Functions](#) for C/C++ and Fortran
- [Java Libraries](#)
- [Python Libraries](#)
- [.NET Libraries](#)
- [COM Objects](#)

Call MATLAB from Another Programming Language

You can call MATLAB from another language using [Engine Applications](#). Using MATLAB engine APIs, call MATLAB functions from your own application. MATLAB has APIs for the following languages.

- [Engine API for C++](#)
- [Engine API for Java](#)
- [Engine API for Python](#)
- [Engine API for C](#)
- [Engine API for Fortran](#)

To create an engine application, install a MATLAB-supported compiler and use the `mex` command to build the application.

Call Your Functions as MATLAB Functions

You can write your own functions and call them as MATLAB functions using [MEX APIs](#). You can write MEX functions in the following languages.

- [C++ MEX APIs](#)
- [C MEX APIs](#)
- [Fortran MEX APIs](#)

To create a MEX file, install a MATLAB-supported compiler and use the `mex` command to build the function.

Communicate with Web Services

You can communicate with web services from MATLAB.

- MATLAB [RESTful web services functions](#) allow non-programmers to access many web services using HTTP GET and POST methods.
- For functionality not supported by the RESTful web services functions, use the [HTTP Interface classes](#) for writing customized web access applications.
- If your web service is based on Web Services Description Language (WSDL) document technologies, then use the MATLAB [WSDL functions](#).

Related Topics

- [External Language Interfaces](#)

External Websites

- [MATLAB Supported Interfaces to Other Languages](#)
 - [Supported and Compatible Compilers](#)
-