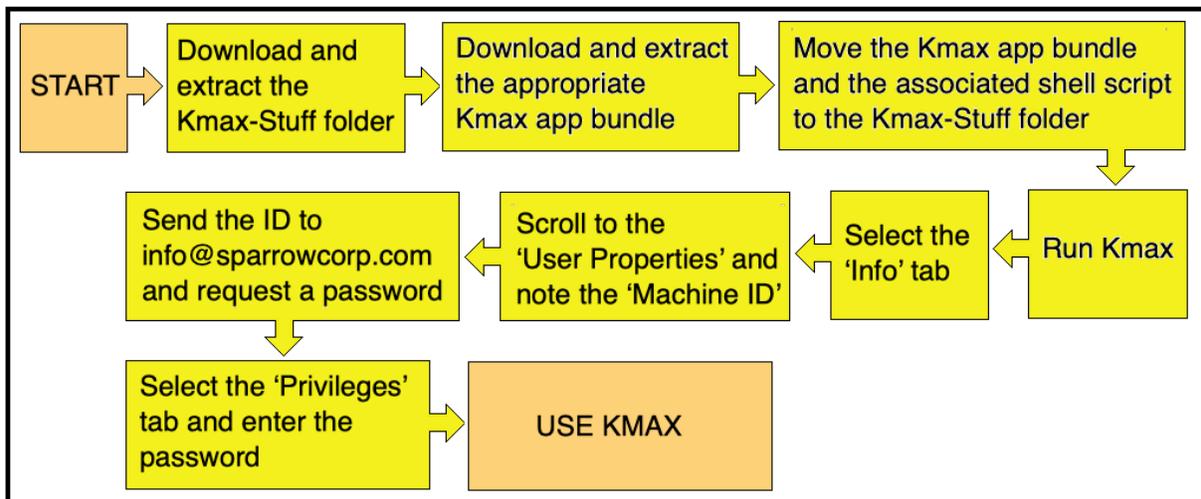




Welcome to Kmax version 11.17

Kmax is a cross-platform, Java-based application that will run on Windows, Linux, or MacOS. This release has been tested on the Windows 10, MacOS 10.15, and Fedora 35, 64-bit platforms. The following flow diagram provides a overview of how to get started with Kmax:



Preparing a Host Computer to Run Kmax

To insure that a computer can run Kmax properly, it should have an appropriate version of the OS you plan to use. Install OS upgrades according to the instructions provided by the vendor.

The platform OS must be compatible with Java and JavaF, versions 17. The current version of the Kmax application is tested on the following platforms:

OS	OS-Version
MacOS	10.15
Linux	Fedora 35
Windows	10

Note that in most cases the user need not install the Java JDK or JavaFX on the host computer since the Kmax application bundle has private copies of the required Java JDK and JavaFX modules. If the user wants to run the Kmax.jar file directly, the JAVA_HOME, KMAX_STUFF and PATH_TO_FX environment variables must be set.

Installing Kmax

The current Kmax version is packaged as an application bundle and has private versions of the Java JDK and JavaFX modules included in the application package. To install Kmax, first download, unzip and move the Kmax-Stuff folder to your desired location. Second, download the app bundle and copy the bundle and the associated script file to the Kmax-Stuff folder.

Kmax-Stuff: Download the Kmax-Stuff zip archive from the Sparrow Corp website (www.sparrowcorp.com) and unzip to an appropriate location. If you are installing a new version over an older copy of Kmax, be careful not to overwrite files in the

Kmax-Stuff folder that were added or edited since the last installation.

App Bundle: Next download the application bundle, either Kmax-Mac-App, Kmax-Linux-App, or Kmax-Win-App, appropriate for your platform and unzip the zip archive. Move the application bundle and the associated script file, if there is one, to the Kmax-Stuff directory.

On Mac computers the bundle is named Kmax-Mac.app and does not require an associated script file. On linux the bundle will appear as a directory and be named Kmax-Linux-App and a separate .sh file is provided to run the bundle. On Windows the bundle will appear as a directory and be named Kmax-Win-App and a separate .bat file is provided to run the bundle.

If the application bundle is executed from inside the Kmax-Stuff directory, Kmax can find drivers and other extensions in the Extensions folder. If Kmax is running from another location, the KMAX_STUFF environment variable should be set so that the Extensions folder can be found.

Running Kmax

On MacOS just double click the Kmax-Mac.app application bundle. If a dialog pops up that says: **“Kmax-Mac.app” is damaged and can’t be opened. You should move it to the Trash**”, it means that MacOS GateKeeper is holding the **Kmax-Mac.app** in quarantine. To release the app from quarantine you need to remove the ‘quarantine’ attribute from the app. Open the terminal app and issue the command: `sudo xattr -r -d`

com.apple.quarantine /path/to/Kmax-Mac.app. To check that the extended attribute has been removed, issue the command: `sudo xattr /path/to/Kmax-Mac.app`.

On Linux the properties of the Kmax-Linux-App.sh file may not allow the file to be executed with a double-click. To cause the system to execute the shell file, open a 'Files' window and locate the Kmax-Linux-App.sh file. From the drop-down context menu of the .sh file, select the Properties item. In the dialog select the 'Permissions' tab, and then select the 'Execute' option. Now you should be able to double-click the Kmax-Linux-App.sh file to start Kmax from the Kmax-Linux-App application bundle. The permissions may also be changed in a terminal window by cd-ing to the Kmax-Stuff folder and entering the 'chmod' command (a means all users):

```
$: chmod a+x Kmax-Linux-App.sh
```

Then just double-click the Kmax-Linux-App.sh file to start Kmax from the application bundle.

On Windows just double-click the Kmax-Win-App.bat file. When the Kmax-Win-App.bat file is first copied to a windows system, execution of the .bat file may be blocked by the Windows OS. Unblock the execution by opening the Kmax-Win-App.bat file properties dialog and clicking the 'Unblock' checkbox. Now you should be able to double-click the Kmax-Win-App.bat file to start Kmax from the Kmax-Win-App application bundle.

Running Kmax From the Command Line

The Kmax application can also be started by executing the Kmax.jar file directly from the command line, but this requires that the proper versions of the Java JDK and JavaFX are properly installed and configured. The environment variables: KMAX_STUFF, PATH_TO_FX, and JAVA_HOME must also be defined to point to the appropriate locations. After this has been done and validated, open a terminal window or shell tool and 'cd' to the Kmax-Stuff folder (e.g. cd ~/Kmax-Stuff) and execute the command:

```
$: java -p $PATH_TO_FX --add-modules=javafx.controls,javafx.swing -jar Kmax.jar
```

- -p is shorthand for --module-path.
- On windows use %PATH_TO_FX% instead of \$PATH_TO_FX.

Executing Kmax from the command line may provide more detailed error messages than when executing Kmax from the application bundle. This can be helpful if you are developing or testing Kmax external libraries or Kmax drivers.

The Kmax Password Dialog

When Kmax executes, the Kmax “Welcome Window” is displayed. Read the license agreement. If you do not accept the

license agreement, quit the application and remove the Kmax software from your system. If you accept the license agreement, select the Privileges tab. From here you may click “Demo” button to run Kmax in demonstration mode. If you have purchased a license or have requested an evaluation license for Kmax, click the “Privileges” tab and note the Machine ID. You will need the Machine ID to request a password from Sparrow. You may also select the “Info” tab and copy the entire contents of that window and send it to Sparrow with your request. This information includes the machine ID and can be used to identify the details of your installation should you require technical support from Sparrow in the future. Send the Machine ID or the “Info” text to support@sparrowcorp.com and request a password. Enter this password in the “Privileges” tab of the “Welcome” window and click ‘Update’ to enable your Kmax license. If you are installing a Kmax upgrade with the same major version and already have a valid license on your computer, you will not need to request another password since the license will still be valid. If you are moving the Kmax application to another computer or to another user account, you will need to run Kmax for the first time, record your new machine ID, and request a new password from SPARROW as described above.

Developing Kmax Tools and Drivers

The Java Development Kit (JDK) is no longer required to run the Kmax application since the Kmax application bundle has private copies of the required Java JDK and JavaFX modules. However, to develop/run Kmax Tools, Kmax external libraries, or Kmax drivers the Java JDK and JavaFX SDK must be

properly installed. Download and install the desired version of the Java JDK (and the JavaFX SDK if needed) and install according to the corresponding instructions.

Set the 'JAVA_HOME' environment variable to point to '.../java/jdk-[ver]' folder. Then if required, download and install the corresponding version of the JavaFX SDK. Set the 'PATH_TO_FX' environment variable to point to '.../javaFX-sdk-[ver]/lib' folder.

On MacOS or Linux the 'JAVA_HOME', 'PATH_TO_FX', and 'KMAX_STUFF' environment variables can be set for the current terminal session by entering the following commands in the terminal window:

```
$: export JAVA_HOME=/path/to/java/jdk-  
[ver]  
$: export PATH_TO_FX=/path/to/  
javafx[ver]/lib  
$: export KMAX_STUFF=/path/to/Kmax-  
Stuff
```

ON Windows the file system does not always handle spaces in file paths in a consistent manner. File path names with spaces may need to be enclosed in quotations. It may also be helpful to use the %ProgramFiles% environment variable to reference the path to the "Program Files" directory to avoid the problem. Environment variables in the Windows OS are not case-sensitive.

Checking for correct versions of java and javac

If you are installing the Java JDK (and possibly JavaFX), you will need to insure that the correct versions are properly installed and visible to the operating systems. To check that Java is properly installed, open a terminal window (or command line tool) and at the user prompt type:

```
$: java -version
```

The system should respond with something similar to:

```
java version "17.0.1" 2021-10-19 LTS
Java(TM) SE Runtime Environment (build
17.0.12-LTS-39)
Java HotSpot(TM) 64-Bit Server VM
(build 17.0.1+12-LTS-39, mixed mode,
sharing)
```

To insure that javac is installed and that it is in the execution path from the command line type:

```
$: javac -version
```

The system should respond with the version information:

```
javac 17.0.1
```

If either of these tests fail, either the java JDK is not properly installed or the environment variables are not properly set up. Ensure that the JAVA_HOME, PATH_TO_FX, KMAX_STUFF and path variables point to the correct locations.

Other required drivers and libraries

The Kmax distribution package includes natively-compiled code modules that Kmax uses to connect computer interfaces to external devices. This native code is platform specific and often depends on libraries or device drivers provided by the manufacturer of the external device. The availability of these modules depends on the quality of support available from the manufacturer of the device, and thus not all devices are supported on all platforms. Please contact the device manufacturer or Sparrow Corporation (support@sparrowcorp.com) to determine if your specific device is supported.

If you plan to use a USB device, the Kmax driver module may require the libusb 1.0 library. Download and install the latest version available for your operating system. Make sure that all path variables are set so that the shared library can be found at runtime. If the OS path variables are not set correctly, unpredictable results may occur.

Installing Kmax Driver Modules

Kmax drivers are jar files containing classes that implement the I/O methods that Kmax uses to communicate with external devices. The drivers are located in architecture-specific folders in the Kmax Extensions folder. The folder hierarchy is designed so that Kmax can locate drivers that are compatible with the architecture of the current platform and OS. If you want to add a driver to the Kmax environment, make sure that the driver files

are placed in the appropriate folder for the system architecture.

Current folders include:

Linux-amd64

MacOS

MacOS-AArch64

Windows-amd64

(DEVELOPER NOTE: AMD64 is the name that AMD gave to their new 64 bit architecture when it was first proposed. It is the 64-bit version of the x86 instruction set and is sometimes referred to as x86-64, x86_64, and x64. The original specification was created by AMD and so we use "-amd64" to label folders that provide 64 bit Kmax drivers. The AMD64 (x86-64) specification is distinct from the Intel Itanium (formerly IA-64) architecture, which is not compatible with the native instruction set of the x86 architecture.)

Kmax Documentation

The Kmax reference manual is provided as a set of HTML and image files and may be read using any web browser. The manual can be accessed from the Kmax Help menu. Keep the HelpFiles folder in the same folder as the Kmax application and bookmark the KmaxManual.html home page in your browser for easy access. While the manual will be a valuable reference, the best way to learn how to design and build toolsheets is to explore the toolsheet examples provided with the Kmax distribution.

Important Compatibility Information

Kmax v10 toolsheets can be opened with Kmax v11 and above

While the v10 toolsheets can be opened, they may require modifications in order to compile and run. Most of these modifications are a result of replacing java swing with the javaFX components. See the 'ReadMe-LegacyToolsheets' file for more information.

Other Known Issues and Problems

If a Kmax Tool does not close properly, it may sometimes leave temporary folders in the same directory as the Tool. After Kmax is closed these folders may be deleted. Do not delete these temporary folders while the Kmax Tool is running.

Slashes or colons in directory or file names may produce unexpected effects, since they are used as file (directory) separators on some platforms. Spaces in file names may also produce unexpected effects.

Double-clicking a toolsheet document (.ktlsh) may NOT open the file with Kmax application.

Check the SparrowCorp website often for the most current

Kmax information.