

LiveCode 9.6.9-rc-1 Release Notes

- [Overview](#)
- [Platform support](#)
 - [Windows](#)
 - [Linux](#)
 - [macOS](#)
 - [iOS](#)
 - [Android](#)
 - [HTML5](#)
- [Setup](#)
 - [Installation](#)
 - [Uninstallation](#)
 - [Reporting installer issues](#)
 - [Activating LiveCode](#)
 - [Command-line installation](#)
 - [Command-line uninstallation](#)
 - [Command-line activation](#)
- [New Features](#)
 - [Standalone Builder](#)
- [Breaking Changes](#)
 - [Database Access](#)
- [Issues Resolved](#)
 - [Features implemented](#)
 - [Bugs fixed](#)
- [Previous release notes](#)

Overview

This document describes all the changes that have been made for LiveCode 9.6.9-rc-1, including bug fixes and new syntax.

Platform support

The engine supports a variety of operating systems and versions. This section describes the platforms that we ensure the engine runs on without issue (although in some cases with reduced functionality).

Windows

LiveCode supports the following versions of Windows:

- Windows 7 (both 32-bit and 64-bit)
- Windows Server 2008
- Windows 8.x (Desktop)
- Windows 10
- Windows 11

Note: On 64-bit Windows installations, LiveCode can run either as a 32-bit application through the WoW layer or as a native 64-bit Windows application, depending on the installer that is chosen.

Linux

LiveCode supports the following Linux distributions, on 32-bit or 64-bit Intel/AMD or compatible processors:

- Ubuntu 14.04 and 16.04
- Fedora 23 & 24
- Debian 7 (Wheezy) and 8 (Jessie) [server]
- CentOS 7 [server]

LiveCode may also run on Linux installations which meet the following requirements:

- Required dependencies for core functionality:
 - glibc 2.13 or later
 - glib 2.0 or later
- Optional requirements for GUI functionality:
 - GTK/GDK 2.24 or later
 - Pango with Xft support
 - esd (optional, needed for audio output)
 - mplayer (optional, needed for media player functionality)
 - lcms (optional, required for color profile support in images)
 - gksu (optional, required for privilege elevation support)

Note: If the optional requirements are not present then LiveCode will still run but the specified features will be disabled.

Note: The requirements for GUI functionality are also required by Firefox and Chrome, so if your Linux distribution runs one of those, it will run LiveCode.

macOS

The macOS engine supports Intel architecture machines running the following operating system versions:

- 10.9.x (Mavericks)
- 10.10.x (Yosemite)

- 10.11.x (El Capitan)
- 10.12.x (Sierra)
- 10.13.x (High Sierra)
- 10.14.x (Mojave)
- 10.15.x (Catalina)
- 11.x (Big Sur)
- 12.x (Monterey)

The macOS engine supports Apple architecture machines running the following operating system versions:

- 11.x (Big Sur)
- 12.x (Monterey)

The macOS IDE supports building macOS standalones with an Apple architecture slice on:

- 10.13.x (High Sierra)
- 10.14.x (Mojave)
- 10.15.x (Catalina)
- 11.x (Big Sur)
- 12.x (Monterey)

When running the IDE in older macOS versions, the request for adding an Apple architecture slice will be ignored.

Note: Apple architecture support is currently experimental. To run the IDE using Apple architecture (on supported machines), you must toggle the `Open using Rosetta` option to off in the `LiveCode.app` bundle's `Get Info` pane in Finder. To build a standalone with native Apple architecture support you must explicitly choose the `macOS Apple` option in standalone settings.

iOS

iOS deployment is possible when running LiveCode IDE on a Mac, and provided Xcode is installed and has been set in LiveCode *Preferences* (in the *Mobile Support* pane).

Currently, the supported versions of Xcode are:

- Xcode 10.1 on MacOS 10.13 (Note: You need to upgrade to 10.13.4)
- Xcode 11.3 on MacOS 10.14 (Note: You need to upgrade to 10.14.4)
- Xcode 12.4 on MacOS 10.15 and above (Note: You need to upgrade to 10.15.4)
- Xcode 13.2 on MacOS 11 and above (Note: You need to upgrade to 11.3+)

It is also possible to set other versions of Xcode, to allow testing on a wider range of iOS simulators. For instance, on MacOS 10.14 (High Sierra), you can add *Xcode 10.1* in the *Mobile Support* preferences, to let you test your stack on the *iOS Simulator 12.1*.

We currently support building against the following versions of the iOS SDK:

- 12.1 (included in Xcode 10.1)
- 13.2 (included in Xcode 11.3)
- 14.4 (included in Xcode 12.4)
- 15.2 (included in Xcode 13.2)

Android

LiveCode allows you to save your stack as an Android application, and also to deploy it on an Android device or simulator from the IDE.

Android deployment is possible from Windows, Linux and Mac OSX.

The Android engine supports devices using x86, x86-64, ARM and ARM64 processors. It will run on the following versions of Android:

- 5.0-5.1 (Lollipop)
- 6.0 (Marshmallow)
- 7.x (Nougat)
- 8.x (Oreo)
- 9.0 (Pie)
- 10.0 (Q)
- 11.0 (R)

To enable deployment to Android devices, you need to download the [Android SDK](#), and then use the 'Android SDK Manager' to install:

- the latest "Android SDK Tools"
- the latest "Android SDK Platform Tools"

You also need to install the Java Development Kit (JDK). On Linux, this usually packaged as "openjdk". LiveCode requires JDK version 1.6 or later.

Once you have set the path of your Android SDK in the "Mobile Support" section of the LiveCode IDE's preferences, you can deploy your stack to Android devices.

Some users have reported successful Android Watch deployment, but it is not officially supported.

HTML5

LiveCode applications can be deployed to run in a web browser, by running the LiveCode engine in JavaScript and using modern HTML5 JavaScript APIs.

HTML5 deployment does not require any additional development tools to be installed.

LiveCode HTML5 standalone applications are currently supported for running in recent versions of [Mozilla Firefox](#), [Google Chrome](#) or [Safari](#). For more information, please see the "HTML5 Deployment" guide in the LiveCode IDE.

Setup

Installation

Each version of LiveCode installs can be installed to its own, separate folder. This allow multiple versions of LiveCode to be installed side-by-side. On Windows (and Linux), each version of LiveCode has its own Start Menu (or application menu) entry. On Mac OS X, each version has its own app bundle.

On Mac OS X, install LiveCode by mounting the `.dmg` file and dragging the app bundle to the `Applications` folder (or any other suitable location).

For Windows and Linux, the default installation locations when installing for "All Users" are:

Platform	Path
Windows	<code><x86 program files folder>/RunRev/LiveCode <version></code>
Linux	<code>/opt/livecode/livecode-<version></code>

The installations when installing for "This User" are:

Platform	Path
Windows	<code><user roaming app data folder>/RunRev/Components/LiveCode <version></code>
Linux	<code>~/.runrev/components/livecode-<version></code>

Note: If installing for "All Users" on Linux, either the **gksu** tool must be available, or you must manually run the LiveCode installer executable as root (e.g. using **sudo** or **su**).

Uninstallation

On Windows, the installer hooks into the standard Windows uninstall mechanism. This is accessible from the "Add or Remove Programs" applet in the windows Control Panel.

On Mac OS X, drag the app bundle to the Trash.

On Linux, LiveCode can be removed using the `setup.x86` or `setup.x86_64` program located in LiveCode's installation directory.

Reporting installer issues

If you find that the installer fails to work for you then please report it using the [LiveCode Quality Control Centre](#) or by emailing support@livecode.com.

Please include the following information in your report:

- Your platform and operating system version
- The location of your home or user folder
- The type of user account you are using (guest, restricted, admin etc.)
- The installer log file.

The installer log file can be located as follows:

Platform	Path
Windows 2000/XP	<code><documents and settings folder>/<user>/Local Settings/</code>
Windows Vista/7	<code><users folder>/<user>/AppData/Local/RunRev/Logs</code>
Linux	<code><home>/.runrev/logs</code>

Activating LiveCode

The licensing system ties your product licenses to a customer account system, meaning that you no longer have to worry about finding a license key after installing a new copy of LiveCode. Instead, you simply have to enter your email address and password that has been registered with our customer account system and your license key will be retrieved automatically.

Alternatively it is possible to activate the product via the use of a specially encrypted license file. These will be available for download from the customer center after logging into your account. This method will allow the product to be installed on machines that do not have access to the internet.

Command-line installation

It is possible to invoke the installer from the command-line on Linux and Windows. When doing command-line installation, no GUI will be displayed. The installation process is controlled by arguments passed to the installer.

Run the installer using a command in the form:

```
<installer> install -ui [OPTION ...]
```

where `<installer>` should be replaced with the path of the installer executable or app (inside the DMG) that has been downloaded. The result of the installation operation will be written to the console.

The installer understands any of the following `OPTION`s:

Option	Description
<code>-allusers</code>	Install the IDE for "All Users". If not specified, LiveCode will be installed for the current user only.
<code>-desktopshortcut</code>	Place a shortcut on the Desktop (Windows-only)
<code>-startmenu</code>	Place shortcuts in the Start Menu (Windows-only)
<code>-location LOCATION</code>	The folder to install into. If not specified, the <code>LOCATION</code> defaults to those described in the "Installation" section above.
<code>-log LOGFILE</code>	The file to which to log installation actions. If not specified, no log is generated.

Note: the command-line installer does not do any authentication. When installing for "All Users", you will need to run the installer command as an administrator.

As the installer is actually a GUI application, it needs to be run slightly differently from other command-line programs.

On Windows, the command is:

```
start /wait <installer> install -ui [OPTION ...]
```

Command-line uninstallation

It is possible to uninstall LiveCode from the command-line on Windows and Linux. When doing command-line uninstallation, no GUI will be displayed.

Run the uninstaller using a command of the form:

```
<uninstaller> uninstall -ui
```

Where `.setup.exe` on Windows, and `.setup.x86` on Linux. This executable, for both of the platforms, is located in the folder where LiveCode is installed.

The result of the uninstallation operation will be written to the console.

Note: the command-line uninstaller does not do any authentication. When removing a version of LiveCode installed for "All Users", you will need to run the uninstaller command as an administrator.

Command-line activation

It is possible to activate an installation of LiveCode for all users by using the command-line. When performing command-line activation, no GUI is displayed. Activation is controlled by passing command-line arguments to LiveCode.

Activate LiveCode using a command of the form:

```
<livecode> activate -file LICENSEFILE -passphrase SECRET
```

where `<livecode>` should be replaced with the path to the LiveCode executable or app that has been previously installed.

This loads license information from the manual activation file `LICENSEFILE`, decrypts it using the given `SECRET` passphrase, and installs a license file for all users of the computer. Manual activation files can be downloaded from the [My Products](#) page in the LiveCode account management site.

It is also possible to deactivate LiveCode with:

```
<livecode> deactivate
```

Since LiveCode is actually a GUI application, it needs to be run slightly differently from other command-line programs.

On Windows, the command is:

```
start /wait <livecode> activate -file LICENSE -passphrase SECRET  
start /wait <livecode> deactivate
```

On Mac OS X, you need to do:

```
<livecode>/Contents/MacOS/LiveCode activate -file LICENSE -passphrase SECRET  
<livecode>/Contents/MacOS/LiveCode deactivate
```

New Features

Standalone Builder

Android audio permissions

Support has been added for two permissions related to audio recording to be requested on apps built for Android in the standalone builder:

- The `Record Audio` option maps to `android.permission.RECORD_AUDIO`
- The `Modify Audio Settings` option maps to `android.permission.MODIFY_AUDIO_SETTINGS`

In particular, the `Record Audio` permission is required in apps which use the camera control to record audio alongside video.

Breaking Changes

Database Access

ODBC driver on Linux now requires unixODBC

The ODBC driver on Linux now requires that the `unixODBC` library be installed on the host machine in order to function.

Previously, the driver included its own version of `libiodbc`, however this has compatibility issues with more recent database vendor's drivers.

ODBC driver on macOS now requires iODBC

The ODBC driver on macOS now requires that a default install of version 3.52.15 of the `iODBC` runtime libraries are present on the host machine in order to function. Installable packages of the required version are available from iodbc.org.

Previously, the driver included its own version of the iODBC client library, however this has compatibility issues when running on Apple architecture macs.

Issues Resolved

Features implemented

- 22670** Options to add audio related permissions to android apps have been added to the standalone builder **9.6.9-rc-1**

Bugs fixed

- 9132** The `Go to handler...` submenu in the `Handlers` menu in the script editor is now sorted according to the `Handler List` sorting option chosen in preferences **9.6.9-rc-1**
- 14022** A new menu item `Report an Issue` has been added to the `Help` menu **9.6.9-rc-1**
- 15558** The dictionary has been updated to clarify the names of the text encodings accepted by the `textEncode` and `textDecode` functions **9.6.9-rc-1**
- 15790** The `Newsletters` item of the `Help` menu has been renamed to `Blog` and now points to the LiveCode blog **9.6.9-rc-1**
- 16298** Modal dialogs will no longer flash briefly before being displayed on Windows **9.6.9-rc-1**
- 21412** Clicking on links to bug reports in the release notes from the documentation viewer will now open them in an external browser **9.6.9-rc-1**
- 22417** Clicking the bottom left of a collapse branch icon in the project browser now works correctly **9.6.9-rc-1**
- 22831** The names of the localization resources used in macOS standalones have been updated to use ISO-639 codes to better support codesigning and notarization **9.6.9-rc-1**
- 23118** The engine will no longer crash when computing the `formattedRect` of a paragraph containing two adjacent VTAB characters **9.6.9-rc-1**
- 23155** The extension builder will no longer fail to compile an extension which has hidden (`.` prefixed) files in its source folder **9.6.9-rc-1**
- 23388** Pressing return in the search bar of the dictionary no longer clears the search bar contents and results **9.6.9-rc-1**
- 23399** The dictionary entry for the `editionType` property has been updated to be inline with the current product licensing options **9.6.9-rc-1**
- 23514** The dictionary now correctly documents the `allowsInlineMediaPlayback` mobile control browser property **9.6.9-rc-1**
- 23516** Creating a new stack will no longer clear the project browser when it was previously displaying deeply nested controls **9.6.9-rc-1**
- 23608** The `System Requirements` section in the user guide has been updated to refer to the associated release notes **9.6.9-rc-1**

23617	The ODBC driver on linux no longer fails when attempting to use an installed Oracle driver	9.6.9-rc-1
23662	The standard PDF icon has been added to the <code>iconsvg</code> library, making it available to the <code>svgpath</code> widget with name <code>pdf</code>	9.6.9-rc-1
23675	The dictionary now correctly documents the <code>CP1252</code> text encoding as being only available on Windows in the <code>textEncode</code> and <code>textDecode</code> functions	9.6.9-rc-1
23716	The dictionary now correctly lists Android as a supported platform on the print-related syntax which works in the context of <code>open printing to pdf</code>	9.6.9-rc-1
23731	The ODBC driver on macOS now functions correctly when running on Apple architecture machines	9.6.9-rc-1
23764	The socket commands now work when using the server engine on macOS	9.6.9-rc-1
23772	The breakpoints displayed in the gutter are now correct after switching to a different tab in the script editor	9.6.9-rc-1
23777	Pending changes to an object by the property inspector are no longer lost when the selected object changes due to pasting new objects	9.6.9-rc-1
23784	The property inspector will no longer allow a stack to be renamed to the same name as another stack	9.6.9-rc-1
23785	The Android engine is now built using version 31 of the Android API	9.6.9-rc-1
23800	The <code>arrayEncode</code> function will no longer fail when given arrays containing non-integer keys	9.6.9-rc-1
23807	The <code>Evaluation Edition</code> glossary entry has been removed as it no longer contains useful information	9.6.9-rc-1
23811	The IDE no longer leaves files related to setting up extensions in the temporary folder after being launched	9.6.9-rc-1
23821	Setting properties on the mobile control browser using <code>mobileControlSet</code> will no longer cause a crash on iOS in certain circumstances	9.6.9-rc-1
23851	The socket commands now work when using the server engine on Linux	9.6.9-rc-1
23858	The international option in the <code>sort</code> command now supports the user's locale being <code>fr_CA</code> (French Canadian)	9.6.9-rc-1

Previous release notes

- [LiveCode 9.6.8 Release Notes](#)
- [LiveCode 9.6.7 Release Notes](#)
- [LiveCode 9.6.6 Release Notes](#)
- [LiveCode 9.6.5 Release Notes](#)
- [LiveCode 9.6.4 Release Notes](#)

- [LiveCode 9.6.3 Release Notes](#)
- [LiveCode 9.6.2 Release Notes](#)
- [LiveCode 9.6.1 Release Notes](#)
- [LiveCode 9.6.0 Release Notes](#)
- [LiveCode 9.5.1 Release Notes](#)
- [LiveCode 9.5.0 Release Notes](#)
- [LiveCode 9.0.5 Release Notes](#)
- [LiveCode 9.0.4 Release Notes](#)
- [LiveCode 9.0.3 Release Notes](#)
- [LiveCode 9.0.2 Release Notes](#)
- [LiveCode 9.0.1 Release Notes](#)
- [LiveCode 9.0.0 Release Notes](#)
- [LiveCode 8.1.10 Release Notes](#)
- [LiveCode 8.1.9 Release Notes](#)
- [LiveCode 8.1.8 Release Notes](#)
- [LiveCode 8.1.7 Release Notes](#)
- [LiveCode 8.1.6 Release Notes](#)
- [LiveCode 8.1.5 Release Notes](#)
- [LiveCode 8.1.4 Release Notes](#)
- [LiveCode 8.1.3 Release Notes](#)
- [LiveCode 8.1.2 Release Notes](#)
- [LiveCode 8.1.1 Release Notes](#)
- [LiveCode 8.1.0 Release Notes](#)
- [LiveCode 8.0.2 Release Notes](#)
- [LiveCode 8.0.1 Release Notes](#)
- [LiveCode 8.0.0 Release Notes](#)
- [LiveCode 7.1.4 Release Notes](#)
- [LiveCode 7.1.3 Release Notes](#)
- [LiveCode 7.1.2 Release Notes](#)
- [LiveCode 7.1.1 Release Notes](#)
- [LiveCode 7.1.0 Release Notes](#)
- [LiveCode 7.0.6 Release Notes](#)
- [LiveCode 7.0.4 Release Notes](#)
- [LiveCode 7.0.3 Release Notes](#)
- [LiveCode 7.0.1 Release Notes](#)
- [LiveCode 7.0.0 Release Notes](#)
- [LiveCode 6.7.11 Release Notes](#)
- [LiveCode 6.7.10 Release Notes](#)
- [LiveCode 6.7.9 Release Notes](#)
- [LiveCode 6.7.8 Release Notes](#)
- [LiveCode 6.7.7 Release Notes](#)
- [LiveCode 6.7.6 Release Notes](#)
- [LiveCode 6.7.4 Release Notes](#)
- [LiveCode 6.7.2 Release Notes](#)
- [LiveCode 6.7.1 Release Notes](#)
- [LiveCode 6.7.0 Release Notes](#)
- [LiveCode 6.6.5 Release Notes](#)
- [LiveCode 6.6.4 Release Notes](#)
- [LiveCode 6.6.3 Release Notes](#)
- [LiveCode 6.6.2 Release Notes](#)

- [LiveCode 6.6.1 Release Notes](#)
- [LiveCode 6.6.0 Release Notes](#)
- [LiveCode 6.5.2 Release Notes](#)
- [LiveCode 6.5.1 Release Notes](#)
- [LiveCode 6.5.0 Release Notes](#)
- [LiveCode 6.1.3 Release Notes](#)
- [LiveCode 6.1.2 Release Notes](#)
- [LiveCode 6.1.1 Release Notes](#)
- [LiveCode 6.1.0 Release Notes](#)
- [LiveCode 6.0.2 Release Notes](#)
- [LiveCode 6.0.1 Release Notes](#)
- [LiveCode 6.0.0 Release Notes](#)