

LiveCode 9.0.0-dp-1 Release Notes

- [Overview](#)
- [Known issues](#)
- [Platform support](#)
 - [Windows](#)
 - [Linux](#)
 - [Mac](#)
 - [iOS](#)
 - [Android](#)
 - [HTML5](#)
- [Setup](#)
 - [Installation](#)
 - [Uninstallation](#)
 - [Reporting installer issues](#)
 - [Activating LiveCode Indy or Business edition](#)
 - [Command-line installation](#)
 - [Command-line uninstallation](#)
 - [Command-line activation for LiveCode Indy or Business edition](#)
- [Engine changes](#)
 - [Improved ul_TraceLocals output \(9.0.0-dp-1\)](#)
 - [libURLSetStatusCallback no longer requires a target object for the message \(9.0.0-dp-1\)](#)
 - [Platform support end-of-life \(9.0.0-dp-1\)](#)
 - [Field tab alignments in htmlText and styledText \(9.0.0-dp-1\)](#)
 - [Fix problems with printing PDFs to some printers \(9.0.0-dp-1\)](#)
 - [Specific engine bug fixes \(9.0.0-dp-1\)](#)
- [IDE changes](#)
 - [Specific IDE bug fixes \(9.0.0-dp-1\)](#)
- [LiveCode Builder changes](#)
 - [LiveCode Builder Documentation](#)
 - [LiveCode Builder Tools](#)
 - [LiveCode Builder Language](#)
 - [Specific LCB bug fixes \(9.0.0-dp-1\)](#)
- [LiveCode extension changes](#)
 - [Specific extension bug fixes \(9.0.0-dp-1\)](#)
- [Previous release notes](#)

Overview

LiveCode 9.0 enables access to libraries and platform APIs written in many other languages thanks to the community-funded 'Infinite LiveCode' project.

Known Issues

- The installer will currently fail if you run it from a network share on Windows. Please copy the installer to a local disk before launching on this platform.
- The browser widget does not work on 32-bit Linux.
- 64-bit standalones for Mac OS X do not have support for audio recording or the revVideoGrabber external.

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

Note: On 64-bit Windows installations, LiveCode runs as a 32-bit application through the WoW layer.

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.

Note: It may be possible to compile and run LiveCode Community for Linux on other architectures but this is not officially supported.

Mac

The Mac engine supports:

- 10.9.x (Mavericks) on Intel
- 10.10.x (Yosemite) on Intel
- 10.11.x (El Capitan) on Intel
- 10.12.x (Sierra) on Intel

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 6.2 on MacOS X 10.9
- Xcode 6.2 and 7.2 on Mac OS X 10.10
- Xcode 8.0 on MacOS X 10.11
- Xcode 8.0 on MacOS 10.12

It is also possible to set other versions of Xcode, to allow testing on a wider range of iOS simulators. For instance, on MacOS 10.12 (Sierra), you can add *Xcode 6.2* in the *Mobile Support* preferences, to let you test your stack on the *iOS Simulator 8.2*.

We currently support deployment for the following versions of iOS:

- 8.2 [simulator]
- 9.2
- 10.0

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 ARMv7 or ARMv8 processors. It will run on the following versions of Android:

- 4.1-4.3 (Jelly Bean)
- 4.4 (KitKat)
- 5.0-5.1 (Lollipop)
- 6.0 (Marshmallow)

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 `/opt/livecode/livecode-<Path>`

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 Indy or Business edition

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 noui [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 noui [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 noui
```

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 for LiveCode Indy or Business edition

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
```

Engine changes

Improved ul_TraceLocals output (9.0.0-dp-1)

The `ul_TraceLocals` function in `revLibURL` has been improved so that it prints out keys and values of the script local arrays in the library. This is useful for troubleshooting `libURL` issues.

`libURLSetStatusCallback` no longer requires a target object for the message (9.0.0-dp-1)

Passing an object reference as a second parameter to `libURLSetStatusCallback` is no longer required. If no object is passed in then the message will be sent to `revLibURL` itself and you can handle the message anywhere in the message path.

Platform support end-of-life (9.0.0-dp-1)

As [announced on the LiveCode blog](#), running LiveCode on the following platforms is no longer officially supported from LiveCode 9.0 onwards:

- Windows XP
- Windows Server 2003
- Windows Vista
- Android Gingerbread (2.3.3-2.3.7)
- Android Ice Cream Sandwich (4.0)
- OS X Snow Leopard (10.6)
- OS X Lion (10.7)
- OS X Mountain Lion (10.8)
- iOS Simulator 6.1
- iOS Simulator 7.1

Field tab alignments in `htmlText` and `styledText` (9.0.0-dp-1)

The **`styledText`** and **`htmlText`** of a field now include tab alignment information. The **`htmlText`** uses a new `tabalign` attribute with a list of alignments, e.g.

```
<p tabalign='left,center,right'>left&09;middle&09;right&09;</p>
```

The **`styledText`** stores tab alignment in a "tabalign" key in each paragraph's "style" array, e.g.

```
get tStyledText[1]["style"]["tabalign"]
```


Fix problems with printing PDFs to some printers (9.0.0-dp-1)

It was possible for LiveCode to generate PDFs which were incompatible with some printers. This has been fixed by upgrading the PDF generation library which LiveCode uses (cairo).

Specific engine bug fixes (9.0.0-dp-1)

- 14645** Field tab alignments in htmlText and styledText
- 14651** There is no documentation entry for "currentcard"
- 15384** Fix incorrect handling of a 204 response from a server.
- 15865** Fixed Dictionary description for "is not among"
- 16211** Fix compilation errors with MacOSX SDK 10.10 and higher
- 16965** Fix incorrect placement of browser widget after stack rect change when fullscreenmode used.
- 17008** Fix selection handles remaining after selected object deleted
- 17247** Remove selection artefacts when handles are drawn outside of parent group rect
- 17541** Fix problems with printing PDFs to some printers
- 17622** Fix extra data added on Windows when pasting html data from the clipboard
- 17779** Fix scrolling group drawing outside its bounds when acceleratedRendering used.
- 18111** Make PDF user guide typography match dictionary view
- 18125** Fix Dictionary example for is within
- 18254** Improve efficiency of equality operators on binary data
- 18293** Crash when deleting a stack that is used as a popup menu
- 18297** Broken references in "filename of stack" dictionary entry
- 18343** Fix incorrect result from itemOffset when first character of stringToSearch is the delimiter
- 18349** Fix variable contents modified when used to set stack name
- 18357** dispatch documentation should mention arguments can be arrays
- 18379** Don't include incorrectly copied resource forks in standalones
- 18406** Fix delay in triggering handlers when called by JavaScript in browser widget
- 18440** Respect SB Copy Files pane relative / absolute path distinction on mobile
- 18444** Make sure put cookie with empty value works as expected
- 18465** Syntax: mouseUp mouseButtonNumber
- 18472** 'load url' is not properly cleaned up on socketError
- 18488** Error returned by hostnametoaddress was not being reported in libURL.
- 18496** Fix memory leak when using filter on unicode strings
- 18498** Ensure bundled Android externals are available on Windows and Linux
- 18521** Resolve folder path before processing files(folder) and folders(folder)
- 18537** Fix crash when saving stack on OSX ElCapitan

- 18566** libURL inserts "://" between host and port when creating CONNECT request
- 18579** Support defaultNetworkInterface for the accept command
- 18588** Fix a crash due to pending messages to deleted objects
- 6506** Fix regression to watching global variables

IDE changes

Specific IDE bug fixes (9.0.0-dp-1)

- 13997** Fix issue creating breakpoints via the new breakpoint dialog
- 15830** Improve user feedback for invalid breakpoint conditions
- 18029** Show 20 fonts at a time in property inspector font menu
- 18043** Add warning about numerical names to user guide.
- 18290** Use datagrid template safely while building custom headers
- 18292** Fix hidden palettes not reappearing
- 18355** Bring script editor and documentation stacks to front if the stack is already open when navigating to content
- 18452** Saving a substack from the Project Browser no longer asks for a path to save
- 18455** Show the correct version of LiveCode in Start Center title
- 18460** Mark stack as edited when property changed from the PI
- 18475** textFont of control does not get set when tabbing out of textFont comboBox in P.I.
- 18483** Prevent iOS display name standalone setting becoming utf-8 encoded data

LiveCode Builder changes

LiveCode Builder Documentation

Style guide

- Updated naming guide for handlers and types
- Added indentation and wrapping guidelines
- New section with widget-specific recommendations

LiveCode Builder Tools

lc-run

- **lc-run** now has the *experimental* ability to load and run bytecode assemblies containing multiple LCB modules. To construct a multi-module bytecode assembly, simply concatenate several `.lcm` module files together. The first module found in a bytecode

assembly is treated as its main module.

lc-compile

Errors

- Parsing of numeric literals, in general, has been tightened up. In particular, the compiler will detect invalid suffixes on numeric literals meaning you cannot accidentally elide a number with an identifier.

```
1.344foo -- ERROR
0xabcdefgh -- ERROR
0b010432 -- ERROR
```

Messages

- Errors, warnings and informational messages now display the affected line of code and visually indicate the position where the problem was found. For example, the output might look like:

```
foo.lcb:2:26: error: Identifier 'InvalidExpression' not declared
constant kBadConstant is InvalidExpression
                        ^
```

LiveCode Builder Language

Literals

- Base 2 (binary) integer literals can now be specified by using a "0b" prefix, e.g.

```
0b0000
0b1010
```

- Base 16 (hexadecimal) integer literals can now be specified by using a "0x" prefix. e.g.

```
0xdeadbeef
0x0123fedc
```

Specific LCB bug fixes (9.0.0-dp-1)

18086 Improve and expand LCB style guide

18385 lc-run: Load multi-module bytecode assemblies.

18463 Show correct error position when source line includes tabs

LiveCode extension changes

Specific extension bug fixes (9.0.0-dp-1)

18319 Prevent segmented control fill from bleeding outside border.

18391 Correctly order default marker styles

Previous 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 6.7.9 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 6.7.11 Release Notes](#)
- [LiveCode 6.7.10 Release Notes](#)
- [LiveCode 6.7.1 Release Notes](#)
- [LiveCode 6.7.0 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](#)

