

LiveCode 6.1.1-rc-4 Release Notes

Table of contents

Overview

Known issues

Platform support

Windows

Linux

Mac

Setup

Installation

Uninstallation

Reporting installer issues

Activation

Multi-user and network install support (4.5.3)

Command-line installation

Command-line activation

Proposed changes

Engine changes

Access to the current behavior object. (experimental)

Added support for reading RLE compressed BMP images

Crash setting location of image with filename set to empty

Find Malfunction in Script Editor for both >*/ and double-quote (")

App store submission warns about lack of push notification entitlement even for apps not using push notifications.

Specific bug fixes (6.1.1-rc-4)

Specific bug fixes (6.1.1-rc-3)

Specific bug fixes (6.1.1-rc-2)

Specific bug fixes (6.1.1-rc-1)

Previous Release Notes

Overview

This document describes all the changes that have been made for LiveCode 6.1.1-rc-4, including bug fixes and new syntax.

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.

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

The engine supports the following Windows OSes:

- Windows 2000 SP4
- Windows XP SP2 and above
- Windows Server 2003
- Windows Vista SP1 and above (both 32-bit and 64-bit)
- Windows 7 (both 32-bit and 64-bit)
- Windows Server 2008

Note: On 64-bit platforms the engine still runs as a 32-bit application through the WoW layer.

Linux

The linux engine requires the following:

- 32-bit installation, or a 64-bit linux distribution that has a 32-bit compatibility layer
- 2.4.x or later kernel
- X11R5 capable Xserver running locally on a 24-bit display
- glibc 2.3.2 or later
- gtk/gdk/glib (optional – required for native theme support)
- pango/xfnt (optional – required for pdf printing, anti-aliased text and unicode font support)
- lcms (optional – required for color profile support in JPEGs and PNGs)
- gksu (optional – required for elevate process support)

Note: The optional requirements (except for gksu and lcms) are also required by Firefox and Chrome, so if your linux distribution runs one of those, it will run the engine.

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

Note: LiveCode and standalones it builds may work on remote Xservers and in other bit-depths, however this mode of operation is not currently supported.

Mac

The Mac engine supports:

- 10.4.11 (Tiger) on Intel and PowerPC
- 10.5.8 and later (Leopard) on Intel and PowerPC
- 10.6.x (Snow Leopard) on Intel
- 10.7.x (Lion) on Intel
- 10.8.x (Mountain Lion) on Intel

Note: *The engine runs as a 32-bit application regardless of the capabilities of the underlying processor.*

Setup

Installation

Each distinct version has its own complete folder – multiple versions will no longer install side-by-side: on Windows (and Linux), each distinct version will gain its own start menu (application menu) entry; on Mac, each distinct version will have its own app bundle.

The default location for the install on the different platforms when installing for 'all users' are:

- Windows: <x86 program files folder>/RunRev/ LiveCode 6.1.1-rc-4
- Linux: /opt/runrev/livecode-6.1.1-rc-4
- Mac: /Applications/ LiveCode 6.1.1-rc-4.app

The default location for the install on the different platforms when installing for 'this user' are:

- Windows: <user roaming app data folder>/RunRev/Components/LiveCode 6.1.1-rc-4
- Linux: ~/.runrev/components/livecode-6.1.1-rc-4
- Mac: ~/Applications/ LiveCode 6.1.1-rc-4.app

Note: *If your linux distribution does not have the necessary support for authentication (gksu) then the installer will run without admin privileges so you will have to manually run it from an admin account to install into a privileged location.*

Uninstallation

On Windows, the installer hooks into the standard Windows uninstall mechanism. This is accessible from the appropriate pane in the control panel.

On Mac, simply drag the app bundle to the Trash.

On Linux, the situation is currently less than ideal:

- open a terminal
- `cd` to the folder containing your rev install. e.g.

```
cd /opt/runrev/livecode-6.1.1-rc-4
```

- execute the `.setup.x86` file. i.e.

```
./ .setup.x86
```

- follow the on-screen instructions.

Reporting installer issues

If you find that the installer fails to work for you then please file a bug report in the RQCC or email support@runrev.com so we can look into the problem.

In the case of failed install it is vitally important that you include the following information:

- Your platform and operating system version
- The location of your home/user folder
- The type of user account you are using (guest, restricted, admin etc.)
- The installer log file located as follows:
 - **Windows 2000/XP:** <documents and settings folder>/<user>/Local Settings/

- **Windows Vista/7:** <users folder>/<user>/AppData/Local/RunRev/Logs
- **Linux:** <home>/<runrev>/logs
- **Mac:** <home>/Library/Application Support/Logs/RunRev

Activation

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.

Multi-user and network install support (4.5.3)

In order to better support institutions needing to both deploy the IDE to many machines and to license them for all users on a given machine, a number of facilities have been added which are accessible by using the command-line.

Note: *These features are intended for use by IT administrators for the purposes of deploying LiveCode in multi-user situations. They are not supported for general use.*

Command-line installation

It is possible to invoke the installer from the command-line on both Mac and Windows. When invoked in this fashion, no GUI will be displayed, configuration being supplied by arguments passed to the installer.

On both platforms, the command is of the following form:

```
<exe> install noui options
```

Here *options* is optional and consists of one or more of the following:

-allusers	Install the IDE for all users. If not specified, the install will be done for the current user only.
-desktopshortcut	Place a shortcut on the Desktop (Windows-only)
-startmenu	Place shortcuts in the Start Menu (Windows-only)
-location <i>location</i>	The location to install into. If not specified, the location defaults to those described in the <i>Layout</i> section above.
-log <i>logfile</i>	A file to place a log of all actions in. If not specified, no log is generated.

Note that the command-line variant of the installer does not do any authentication. Thus, if you wish to install to an admin-only location you will need to be running as administrator before executing the command. As the installer is actually a GUI application, it needs to be run slightly differently from other command-line programs.

In what follows <installerexe> should be replaced with the path of the installer executable or app (inside the DMG) that has been downloaded.

On Windows, you need to do:

```
start /wait <installerexe> install noui options
```

On Mac, you need to do:

```
"<installerexe>/Contents/MacOS/installer" install noui options
```

On both platforms, the result of the installation will be written to the console.

Command-line activation

In a similar vein to installation, it is possible to activate an installation of LiveCode for all-users of that machine by using the command-line. When invoked in this fashion, no GUI will be displayed, activation being controlled by any arguments passed.

On both platforms, the command is of the form:

```
<exe> activate -file license -passphrase phrase
```

This command will load the manual activation file from *license*, decrypt it using the given *passphrase* and then install a license file for all users of the computer. Manual activation files can be downloaded from the 'My Products' section of the RunRev customer accounts area.

This action can be undone using the following command:

```
<exe> deactivate
```

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

In what follows <livecodeexe> should be replaced with the path to the installed LiveCode executable or app that has been previously installed.

On Windows, you need to do:

```
start /wait <livecodeexe> activate -file license -passphrase phrase
start /wait <livecodeexe> deactivate
```

On Mac, you need to do:

```
"<livecodeexe>/Contents/MacOS/LiveCode" activate -file license -passphrase phrase
"<livecodeexe>/Contents/MacOS/LiveCode" deactivate
```

On both platforms, the result of the activation will be written to the console.

Proposed changes

The following changes are likely to occur in the next or subsequent non-maintenance release:

- The engine (both IDE and standalone) **will require** gtk, gdk, glib, pango and xft on Linux

Engine changes

Access to the current behavior object. (6.1.1-rc-4 - experimental)

It is now possible to use 'this me' to refer to the object to which the currently executing script is attached. This is particularly useful in the context of chained behaviors, in which context 'this me' will be the object of the current behavior. This differs from 'me' in this context, which will always be the object the behavior is acting upon.

For example, if Button A's behavior property is set to Button B, and Button B has script:

```
on mouseUp
  put the short name of this me
end mouseUp
```

Then clicking on Button A will result in 'B' being output (as the mouseUp handler is in Button B's script).

Note: 'this me' is provisional syntax, it will change in the next major version so scripts relying on it will have to be updated.

Important: This feature is currently experimental. This means that it may not be complete, or may fail in some circumstances that you would expect it to work. Please do not be afraid to try it out as we need feedback to develop it further.

Added support for reading RLE compressed BMP images (6.1.1-rc-3)

Crash setting location of image with filename set to empty (6.1.1-rc-2)

Find Malfunction in Script Editor for both >/ and double-quote (") (6.1.1-rc-2)

The find and replace was always using wildcards() even when Use Regular Expressions was false. The find stack has been updated to allow users to choose the type of search from Regular Expressions, Wildcards or Plain Text.

App store submission warns about lack of push notification entitlement even for apps not using push notifications. (6.1.1-rc-2)

Specific bug fixes (6.1.1-rc-4)

(bug fixes specific to the current build are highlighted in bold, reverted bug fixes are stricken through)

- 11129 **Crash when fetching the encoding or the flaggedRanges property from the templateField.**
- 11126 **Intermittant crashes whilst using the project browser.**

Specific bug fixes (6.1.1-rc-3)

- 11122 The script editor erroneously starts a comment in \$#.
- 11108 Crash in some circumstances when setting the hidden of a paragraph to false.
- 11105 Plugin settings refers to Revolution. This has been updated to refer to LiveCode.
- 11104 Crash when getting childControllDs/Names of group with no controls
- 11103 Push notifications not handled when app is paused
- 10826 Menu Text > Font > Size > Other... does nothing. Now asks the user to input a text size.

- 10689 'the selectedChunk' returns a field reference if the field is embedded in a combo-box.
- 10665 Save this stack as syntax broken in the message box
- 10655 Dragging a file over a script without dropping causes the caret to continue to move in field after drag has ended.
- 10654 Cells in field table mode can become misaligned if there is a change in style just after a tab.
- 10616 Different padding between lines causes hgrid to render incorrectly.
- 10445 Sort international does not work on Android

Specific bug fixes (6.1.1-rc-2)

- 11089 The 'shell()' function on server does not perform end-of-line conversion.
- 11084 Conditional messages such as resizeMode are not searched for in ancestor behaviors.
- 11081 Dropper tool does not work on OSX
- 11080 Android input field fontSize setting value in incorrect units
- 11073 Transparency lost when saving RLE encoded images with LiveCode
- 11064 LiveCode IDE breaks when opening a particular PDF file - uncaught exception fetching the properties of a DataGrid
- 11044 Divide by zero when calculating scrollbar thumbposition
- 11036 IDE no longer exposes an AppleScript dictionary.
- 11025 The PDF Printing external is not being included in Windows Standalones.
- 11020 Setting or getting custom properties from global scope crashes server.
- 11017 Handlers in library stacks and chained behaviors are cannot be executed from the message box.
- 10972 Text does not paste from PDFs loaded into Adobe Reader 9 on Mac.
- 10957 The 'flaggedRanges' property can sometimes report the wrong values in a field containing mixed unicode and non-unicode text.
- 10932 Setting the flaggedRanges of a field containing Unicode text can be very slow.
- 10925 Using add/subtract/multiply/divide on a variable that has not been declared causes subexpressions to be evaluated twice.
- 10902 Selecting regular polygon graphic fails to update title of Property Inspector
- 10893 Copying HTML from clipboard into a text field generates misformatted output
- 10883 Nested groups can return True in sharedBehavior getter
- 10868 Mobile stack saved as iOS retains iOS settings on Windows. As the option is disabled and cannot be unchecked on Windows the standalone builder will disregard this choice on Windows. Selecting another platform to build for will still automatically unselect iOS.
- 10865 Setting secureMode to true prevents an app from handling 'Quit' and 'Preferences' items from the system menu on Mac.
- 10840 When rendering the caret against transparency
- 10800 Get URL on mobile can fail if URL contains whitespace chars
- 10774 LiveCode engine crashes in filter in repeat loop
- 10671 Combining Drag & Drop with "grab me" causes LC IDE to crash

Specific bug fixes (6.1.1-rc-1)

- 11061 Engine crash getting extents of numeric array with 8-digit key
- 11054 Inconsistent Ask/Answer fonts/text.

- 11043 The Project Browser does not update when a group is ungrouped.
- 11037 Crash setting a behavior to an object that would cause a loop in the hierarchy.
- 11024 Cannot drag cards in the Project Browser.
- 11023 About menu moved on mac to help menu.
- 11018 LiveCode crashes opening rotated image with invalid filename
- 11011 app identifier field in standalone applications settings on iOS and Android allows return characters.
- 10971 Strange instabilities in LiveCode IDE with certain stacks.
- 9836 Set cursor to none does not work

Previous Release Notes

- 6.1.0 Release Notes http://downloads.livecode.com/livecode/6_1_0/LiveCodeNotes-6_1_0.pdf
- 6.0.2 Release Notes http://downloads.livecode.com/livecode/6_0_2/LiveCodeNotes-6_0_2.pdf
- 6.0.1 Release Notes http://downloads.livecode.com/livecode/6_0_1/LiveCodeNotes-6_0_1.pdf
- 6.0.0 Release Notes http://downloads.livecode.com/livecode/6_0_0/LiveCodeNotes-6_0_0.pdf