LiveCode 6.1.0 Release Notes

Table of contents

Overview

This document describes all the changes that have been made for LiveCode 6.1.0, 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/xft (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 6.1.0
- Linux: /opt/runrev/livecode-6.1.0
- Mac: /Applications/ LiveCode 6.1.0.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 6.1.0
- Linux: ~/.runrev/components/livecode-6.1.0
- Mac: ~/Applications/ LiveCode 6.1.0.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-6.1.0
```

- 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 *location*	The location to install into. If not specified, the location defaults to those described in the *Layout* section above.
-log *logfile*	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

Crash when using 'flip' on a referenced image. (6.1.0-rc-3)

Previously attempting to use the 'flip' command on a referenced image (image with filename) would cause a crash. Attempting to use the flip command in this way now will cause an error 'object is not an editable image'.

Note: Prior to 6.0, the flip command would work on a referenced image but only temporarily. The new behavior is more consistent with the other image editing operations (such as rotate) and the ability to transform referenced images will be introduced in a subsequent version.

Images with filenames that look like URLs can cause slowdown. (6.1.0-rc-2)

Previously, if an image was set with a filename that roughly looked like a URL, it would cause an attempt to load the URL repeatedly. This could cause slowdown or, in the worse case, an infinite recursive loop. Now, the engine checks filenames more closely to see if they really could be URLs before attempting a fetch, and if a fetch is attempted it will only occur once and not retried if it fails.

ASCII string check (6.1.0-rc-1)

There is a new operator **is an ascii string**. This operator can be used to determine if all characters in the string are ASCII, that is to say that that charToNum() of each character is less than 128.

UUID Generation (6.1.0-dp-2)

There is a new function **uuid()** which can be used to generate UUIDs on all platforms:

    uuid([*type*, [*namespace_id*, *name*]])

If no parameters are specified, or *type* is "random" then it returns a version 4 (random) UUID. A cryptographic quality pseudo-random number generator is used to generate the randomness.

If *type* is "md5" or "sha1" then it returns a version 3 (md5) or version 5 (sha1) UUID. Here *namespace_id* should be the UUID of the namespace in which *name* sits, and *name* can be any string.

Improved properties Property (6.1.0-dp-2)

The 'properties' object property has been significantly revised. In particular on fetching:

- It returns the minimal set of properties of an object to allow it to be recreated exactly.
- It returns unicode variants of properties if, and only if, they are needed.

And on storing it will ensure that properties are set in the correct order to ensure correct recreation of the object.
Note: When setting properties which contain both non-empty color and pattern properties of the same type, the color property will take precedence.

Additional statistical functions (6.1.0-dp-2)

The following statistical functions were added to the engine:

- 'averageDeviation' (or shorter: 'avgDev')
- 'populationStandardDeviation' (or shorter: 'popStdDev')
- 'populationVariance' (or shorter: 'popVariance')
- 'sampleVariance' (or shorter: 'sampVariance' or 'variance')
- 'geometricMean'
- 'harmonicMean'

The following synonyms for existing statistical functions were added to clarify their meaning:

- 'arithmeticMean' and 'mean' as synonyms for 'average'
- 'sampleStandardDeviation' and 'sampStdDev' for 'stdDev'

For a variable

```
theNumbers
```

containing a series of numbers:

```
2,4,4,4,5,5,7,9
```

these statistical functions return the following values:

- **averageDeviation(theNumbers)** ->

  ```
  1.5
  ```

- **populationStandardDeviation(theNumbers)** ->

  ```
  2
  ```

- **sampleStandardDeviation(theNumbers)** ->

  ```
  2.13809
  ```

- **populationVariance(theNumbers)** ->

  ```
  4
  ```

- **sampleVariance(theNumbers)** ->

  ```
  4.571429
  ```

- **arithmeticMean(theNumbers)** ->

  ```
  5
  ```

- **geometricMean(theNumbers)** ->

  ```
  4.603216
  ```

- **harmonicMean(theNumbers)** ->

  ```
  4.201751
  ```

Activation removed in Community edition (6.1.0-dp-2)

We have removed all activation requirements in LiveCode Community edition to better match the spirit of

the open source endeavour. We have instead added an optional signup page on first run, allowing users to create a 'LiveCode account'. This step can be skipped.

As the website evolves further, this account will grant access to all services web services requiring signing throughout the entire LiveCode ecosystem. (Website, forums, samples, commenting, â€¦).

Updated Auto-updater

The auto-updater can now to launched manually from the LiveCode help menu.

The cardIds of group (6.1.0)

A new property **the cardIds** of groups has been added. This returns a list of card ids corresponding to the cards the group is currently placed on. This has the same functionality as the cardNames of group, except it returns ids, not names.

The childControlNames and childControlIds of group (6.1.0)

Two new properties **the childControlNames** and **the childControlsIds** have been added to the group object. These properties return a list of names (or ids) of all the direct children of the group. (This is in contrast to the controlNames and controlIds properties which recurse into subgroups).

the effective visible (6.1.0)

You can now use the 'effective' adjective when getting the visible property of a control. This allows you to determine whether the given control will actually be visible if the card it is on is current and its stack is visible. Specifically, 'the effective visible' returns true if, and only if, if it visible and all parent groups are visible.

Chained Behaviors (6.1.0)

The behavior property of a control currently being used as a behavior will now be taken into account and result in the child behaviour deferring to the parent behavior in the same way a control defers to its behavior. For example, let's say you have the following setup:

```
field "Action" - behaviour set to button "Derived"

button "Derived" - behaviour set to button "Root"

button "Root"
```

Then the message path will be:

```
field "Action"

   button "Derived"

   button "Root"

 <parent of field>
```

iOS Remote Control Support (6.1.0)

Support has been added allowing access to the 'remote control' feature on iOS. This feature allows access to the audio controls both on an external device, and also on the device itself when in lock screen (via double-tapping on the Home button).

To use the new features, remote control access must be first enabled (**iphoneEnableRemoteControl**) and then the app must play audio. You also need to ensure that the audio category is set to 'playback' (**iphoneSetAudioCategory "playback"**).

The remote controls will be delivered via an **remoteControlReceived** message.

The information to display can be configured using **iphoneSetRemoteControlDisplay**.

Enabling and Disabling Remote Control

> **iphoneEnableRemoteControl**
> **iphoneDisableRemoteControl**
> **iphoneRemoteControlEnabled()**

Use the command **iphoneEnableRemoteControl** to start receiving remote control events after you start playing audio and the command **iphoneDisableRemoteControl** to stop receiving them. The **iphoneRemoteControlEnabled()** function returns true if they are currently enabled.

Receiving Remote Control Events

If remote control is enabled, and a remote control operation occurs then a **remoteControlReceived** message will be sent to the current card.

> **remoteControlReceived** *type*

Here *type* is one of: *play, pause, stop, toggle play pause, next track, previous track, begin seeking forward, begin seeking backward, end seeking forward, end seeking backward*.

Configuring Remote Control Display

When remote controls are enabled, information can be set that may be displayed on the remote control device, or on the (locked) home screen controls.

To configure this information use the **iphoneSetRemoteControlDisplay** command:

> **iphoneSetRemoteControlDisplay** *metadata*

Here, *metadata* is an array with one or more of the following keys:

- title - string
- artist - string
- artwork - either the text of an image or an image filename
- composer - string
- genre - string
- album title - string
- album track count - number
- album track number - number
- disc count - number
- disc number - number
- chapter count - number
- chapter number - number

- playback duration - number
- elapsed playback time - number
- playback rate - number
- playback queue index - number
- playback queue count - number

Note: There is no guarantee that any of this information will be used, it is up to the remote control device / iOS to decide.

Note: This functionality is only available on iOS 5.x and above.

Locking Group Updates (6.1.0)

When a control within a group is resized or moved it causes all parents to recalculate their bounds appropriately. This action can now be temporarily suspended by using 'the lockUpdates' property of the group.

Typically, you'll want to set the lockUpdates of a group to true before performing moving or resizing child controls, and then turn it back to false afterwards. e.g.

```
on updateGroupedControls

  set the lockUpdates of me to true

  set the left of button 1 of me to the left of me + 10

  set the bottom of button 2 of me to the bottom of me - 10

  set the lockUpdates of me to false

end updateGroupedControls
```

Note: The lockUpdates property does not nest, and the group will not resize itself appropriately based on the child controls until it is set back to false.

Random Number Generation (6.1.0)

The **randomBytes()** function has been updated to use the platform-provided cryptographic-quality pseudo-number-generator in the case that the SSL libraries are not available. This change means that this function is now always available on all platforms.

Getting the Page Ranges of a Field (6.1.0)

There is a new property **the pageRanges of <field>**. This property (notionally) splits up the field content into pages based on the height of the field, and then returns a return-delimited list of char ranges. Each char range corresponds to an individual page.

For example, for a field capable of displaying two lines of text and containing these lines:

```
Line 1

Line 2

Line 3
```

```
Line 4

Line 5

Line 6
```

The **pageRanges** would return:

```
1,14

15,28

29,42
```

Specific bug fixes (6.1.0-rc-3)

11006	Crash on Mac when attempting to load a stack with non-ascii title after transferring from Windows.
10989	mobilePickPhoto ignores size parameters if they are passed as strings and not numbers.

Specific bug fixes (6.1.0-rc-2)

10991	If a memory warning is received whilst acceleratedRendering mode is enabled
10990	Memory leak when using acceleratedRendering on iOS in certain cases.
10986	'select before/after field ...' selects the field object
10982	RevDB doesn't work with LiveCode server on Mac
10914	PIE binaries are now built for ARMv7+ meaning the 'Non-PIE Binary' warning from iTunes Connect should no longer occur.

Specific bug fixes (6.1.0-rc-1)

10976	Ungrouping objects can result in their id's changing.
10975	Image objects don't properly resolve filenames to be absolute before searching the cache.
10974	Instability in acceleratedRendering caused in specific circumstances when a static layer moves just above a dynamic layer.
10967	Import snapshot on 16-bit displays on Windows causes a blank image to be produced.

Specific bug fixes (6.1.0-dp-2)

10966	Fetching properties of function expressions now works correctly (e.g. the name of controlAtLoc(the mouseLoc)).
10961	Passing script locals as parameters to external functions doesn't work correctly in behaviors.
10960	Getting 'the lockUpdates of ' now works correctly.
10950	'is a color' now checks the input string more carefully

Specific bug fixes (6.1.0)
(bug fixes specific to the current build are highlighted in bold, reverted bug fixes are stricken through)

10945	**mobileControls() returns deleted controls in the list**

10926   'filter' does not honour the lineDelimiter property.

10863   Group command doesn't throw an error when attempting to group a shared group.

Dictionary additions

- **iphoneDisableRemoteControl** (*command*) has been added to the dictionary.
- **iphoneEnableRemoteControl** (*command*) has been added to the dictionary.
- **iphoneSetRemoteControlDisplay** (*command*) has been added to the dictionary.
- **averageDeviation** (*function*) has been added to the dictionary.
- **geometricMean** (*function*) has been added to the dictionary.
- **harmonicMean** (*function*) has been added to the dictionary.
- **iphoneIdentifierForVendor** (*function*) has been added to the dictionary.
- **iphoneRemoteControlEnabled** (*function*) has been added to the dictionary.
- **populationStandardDeviation** (*function*) has been added to the dictionary.
- **populationVariance** (*function*) has been added to the dictionary.
- **uuid** (*function*) has been added to the dictionary.
- **variance** (*function*) has been added to the dictionary.
- **remoteControlReceived** (*function*) has been added to the dictionary.
- **childControlIDs** (*property*) has been added to the dictionary.
- **childControlNames** (*property*) has been added to the dictionary.
- **pageRanges** (*property*) has been added to the dictionary.

Dictionary changes

- The entry for **combine** (*command*) has been updated.
- The entry for **do** (*command*) has been updated.
- The entry for **export snapshot** (*command*) has been updated.
- The entry for **import snapshot** (*command*) has been updated.
- The entry for **iphoneSetReachabilityTarget** (*command*) has been updated.
- The entry for **mobileClearTouches** (*command*) has been updated.
- The entry for **open process** (*command*) has been updated.
- The entry for **revBrowserSet** (*command*) has been updated.
- The entry for **split** (*command*) has been updated.
- The entry for **average** (*function*) has been updated.
- The entry for **controlAtLoc** (*function*) has been updated.
- The entry for **controlAtScreenLoc** (*function*) has been updated.
- The entry for **iphoneReachabilityTarget** (*command*) has been updated.
- The entry for **iphoneSystemIdentifier** (*function*) has been updated.
- The entry for **listRegistry** (*function*) has been updated.
- The entry for **revXMLChildNames** (*function*) has been updated.
- The entry for **revXMLFirstChild** (*function*) has been updated.
- The entry for **revXMLNextSibling** (*function*) has been updated.
- The entry for **revXMLPreviousSibling** (*function*) has been updated.
- The entry for **standardDeviation** (*function*) has been updated.
- The entry for **adExpandEnd** (*message*) has been updated.
- The entry for **adExpandStart** (*message*) has been updated.
- The entry for **adResizeEnd** (*message*) has been updated.
- The entry for **adResizeStart** (*message*) has been updated.

- The entry for **objectSelectionEnded** (*message*) has been updated.
- The entry for **objectSelectionStarted** (*message*) has been updated.
- The entry for **reachabilityChanged** (*message*) has been updated.
- The entry for **=** (*operator*) has been updated.
- The entry for **is a** (*operator*) has been updated.
- The entry for **is not a** (*operator*) has been updated.
- The entry for **HTMLText** (*property*) has been updated.
- The entry for **RTFText** (*property*) has been updated.
- The entry for **allowDatagramBroadcasts** (*property*) has been updated.
- The entry for **behavior** (*property*) has been updated.
- The entry for **cardIDs** (*property*) has been updated.
- The entry for **formattedText** (*property*) has been updated.
- The entry for **icon** (*property*) has been updated.
- The entry for **invisible** (*property*) has been updated.
- The entry for **listIndex** (*property*) has been updated.
- The entry for **metadata** (*property*) has been updated.
- The entry for **properties** (*property*) has been updated.
- The entry for **rectangle** (*property*) has been updated.
- The entry for **recursionLimit** (*property*) has been updated.
- The entry for **sslcertificates** (*property*) has been updated.
- The entry for **stackLimit** (*property*) has been updated.
- The entry for **textStyle** (*property*) has been updated.
- The entry for **visible** (*property*) has been updated.

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