

LiveCode 4.5.3 Release Notes

Table of Contents

Overview.....	2
Known issues.....	3
Platform support.....	3
Windows.....	3
Linux.....	3
Mac.....	4
Proposed changes.....	4
Setup.....	4
Installation.....	4
Uninstallation.....	5
Reporting installer issues.....	5
Activation.....	5
Multi-user and network install support (4.5.3).....	6
Command-line installation.....	6
Command-line activation.....	6
Engine changes.....	7
Non-blocking DNS resolution.....	7
Improved cursor support.....	7
Improved image export.....	8
BMP Export.....	8
Palette reduction.....	8
Raw data export (experimental).....	8
Color profile support.....	9
Print to pdf support.....	9
Print loop integration.....	10
Options.....	10
Hyperlink support.....	10
Runtime dependency.....	10
Elevated process support (experimental).....	11
Slave process improvements.....	11
Videograbber improvements.....	12
Status icon support (experimental).....	12
Listing sub-keys in the registry (experimental).....	13
Cryptographic-quality random data (experimental).....	13
SHA-1 digest support (experimental).....	13
Public key encryption support (experimental).....	14
Generating key pairs.....	14
Command variants.....	14
Encrypting longer messages.....	14
HTTPS – automatic root certificate discovery (experimental).....	15
Runtime execution stack configuration (4.5.1).....	15
Keyboard layout syncing (4.5.1).....	15
Standalone allowInterrupts default (4.5.1).....	16

MySQL driver improvements (4.5.1).....	16
XML Namespace support (4.5.1).....	16
revBrowser improvements (4.5.1).....	17
Out-of-bounds group scrolling (4.5.3 – experimental).....	17
Linux – font support improvements.....	18
Legacy font support.....	18
Linux – theme improvements.....	18
Linux – ssl improvements.....	18
Runtime dependency.....	18
Mac – bundle changes.....	19
Mac – dock icon support (experimental).....	19
Choosing an image.....	19
Configuring the dock icon menu.....	19
Mac – large progress bar (4.5.3).....	20
Mac – window modification indicator (4.5.3 – experimental).....	20
Windows – ssl improvements (4.5.1).....	20
Maintenance.....	20
Noteworthy changes.....	20
Specific bug fixes (4.5.3).....	21
Specific bug fixes (4.5.2).....	22
Specific bug fixes (4.5.1).....	22
Specific bug fixes (4.5).....	23
IDE changes.....	25
Getting folder locations within the IDE.....	25
Update checker.....	25
Affiliate registration.....	25
Database query builder.....	25
iOS Support (4.5.2).....	26
Access to resources folder (4.5.2).....	26
Standalone builder.....	26
MacOS Classic.....	26
Windows – U3 Support.....	26
Windows – UAC Manifest.....	26
Web.....	27
iOS (4.5.2).....	27
Datagrid.....	27
Maintenance.....	29
Noteworthy changes.....	29
Specific bug fixes (4.5.3).....	30
Specific bug fixes (4.5.2).....	30
Specific bug fixes (4.5.1).....	30
Specific bug fixes (4.5).....	31
Revisions.....	32

Overview

LiveCode 4.5 is a major new release containing a number of new features together with a streamlined install process and improved web-plugin.

This document describes all the changes that have been made – including bug fixes and new syntax. At the time of writing, this information has yet to be integrated into the dictionary or User's Guide.

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 Oracle dbdriver for Mac (PowerPC) is not present in this build.

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:

- 10.3.9 (Panther) on PowerPC
- 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

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

Proposed changes

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

- The engine (both IDE and standalone) **will no longer support** Mac OS 10.3.9 (Panther)
- The engine (both IDE and standalone) **will require** gtk, gdk, glib, pango and xft on Linux

Setup

Installation

The structure of the IDE install has changed significantly in this release.

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 4.5.3
- Linux: /opt/runrev/livecode-4.5.3
- Mac: /Applications/ LiveCode 4.5.3.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 4.5.3
- Linux: ~/.runrev/components/livecode-4.5.3
- Mac: ~/Applications/ LiveCode 4.5.3.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-4.5.3
```
- 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 license system has been replaced in this release.

The new 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 now 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 now 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.

Engine changes

Non-blocking DNS resolution

The **open socket** command no longer blocks on DNS resolution. Instead, if resolution is required the command will return immediately and the DNS lookup will happen in the background. If resolution fails, then a **socketError** message is sent in the same way as if connection fails.

For applications using **hostnameToAddress** directly, its syntax has been augmented:

```
hostnameToAddress(hostname, [ callback ])
```

If the *callback* parameter is specified then the call will return immediately and upon completion of the lookup, the callback will be invoked with the resolved address as a parameter.

Improved cursor support

Cursor support has been improved in several ways.

The image will now automatically process any image to convert to a form suitable for display on the current platform and screen depth taking this burden off the developer. Specifically, the engine will scale the image down to the appropriate size, and reduce the number of colors to the appropriate number of colors.

Additionally, the engine has been updated to take advantage of support for larger and potentially alpha-blended cursors on platforms that support this. Specifically:

- Windows XP and above support full alpha-blended cursors up to 64x64
- More recent Linux distributions support alpha-blended cursors up to a size of 64x64
- Mac supports alpha-blended cursors up to 256x256

The engine now picks up the GTK cursor theme on Linux. (4.5.1)

Note: It appears that alpha-blending support depends on the current screen depth on some platforms.

Improved image export

BMP Export

The export command can now produce Windows BMP format images in the same way as it previously did for gif and png:

export target as bmp

Palette reduction

The **export** command can now perform color reduction. To support this, the following new forms have been added:

export target as (gif | png | bmp) with palette colors

export target as (gif | png | bmp) with (standard | optimized) palette

export target as (gif | png | bmp) with count color optimized palette

The first form allows you to specify a list of up to 256 colors to use in the final palette.

The second form will use either the standard 'websafe' palette, or will compute an optimized palette with at most 256 colors.

The third form allows you to choose the size of the optimized palette to be generated. The number of colors can be at most 256.

If the image has any sort of transparency, then this will utilize one palette entry. (In particular if you ask for a 256 color palette with a transparent image, you may get only 255 colors).

In all cases, optional dithering will be performed as determined by its **dontDither** property.

Note: It is a (relatively) slow process to compute an optimized palette and then remap an image against it – it should not be considered a real-time operation except for very small images.

Note: There is no support for exporting a JPEG with a reduce palette as this format is for continuous-tone images and as such the notion of palette makes no sense.

Raw data export (experimental)

It is possible to export raw image data using the following forms:

export target as raw with palette colors

export target as raw with (standard | optimized) palette

export target as raw with count color optimized palette

export target as raw [argb | bgra | abgr | rgab]

The first three of these operate in the same way as for the other formats as described above except that instead of formatted image data you get the raw palette indices packed appropriately depending on the size of the palette:

<= 2 colors will be 1 bpp

<= 4 colors will be 2 bpp

<= 16 colors will be 4 bpp

<= 256 colors will be 8 bpp

The final form allows export of the full 32-bit data of the image with 8 bits per component. In this case, the components are not pre-multiplied with any alpha channel, and appear ordered in memory in increasing bytes.

e.g. The argb form will give you:

byte 0 = alpha

byte 1 = red

byte 2 = green

byte 3 = blue

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.

Color profile support

The engine will now attempt to use any embedded ICC color profile information that is present in JPEGs and PNGs.

If a JPEG contains a color profile then the engine will attempt to use it to translate the image's colors to the default screen color space before display.

When printing JPEGs, the original JPEG data will now be sent to the printer directly whenever possible. In particular, this means that any intermediate color matching for the screen that the engine performs does not affect the printed output.

As an additional side-effect of color profile support, the engine now also supports YCCK and CMYK JPEG images.

Note: *Support for color profiles depends on the OS - in particular Windows XP and earlier only support ICC v2 profiles, Vista onwards supports ICC v4.*

Print to pdf support

The engine printing system has been augmented with the ability to *print to pdf*. This feature uses the existing print-loop model and virtually no code changes are needed to make existing code use it.

Print loop integration

To start a print loop that outputs directly to a pdf rather than the currently configured printer, use the following form:

open printing to pdf *filename* [**with options** *optionArray*]

This should be performed *instead of* the usual **open printing** command.

Note: *It is not necessary (and unwise!) to set the **formatForPrinting** option on stacks which are being printed to PDF.*

Options

The options array which can be optionally specified when opening a print loop for pdf allows you to add entries to the resulting PDF's *Document Information Dictionary*.

The following keys are supported:

Title, Author, Subject, Keywords, Creator, Producer

Their values can be any string.

Hyperlink support

When inside a pdf print loop you can use additional commands to define hyperlinks and their targets.

To define a target for an internal hyperlink, use the following form:

print anchor *name* **at** *anchorPoint*

Here *name* is used to identify the anchor in related **print link** commands, and *anchorPoint* is the location on the current page to which any such link should jump to.

To define a hyperlink use the following form:

print link to *target* **with rect** *linkRectangle*

If *target* is a url, then the given rectangle will jump to that url when it is clicked. If *target* is not a url, it is assumed to be the name of an internal anchor as specified by the **print anchor** command and clicking in the given rectangle will jump there instead.

When printing fields, any text that has its **linkText** property set will be treated as if a **print link** command had been executed with the contents of the property as target, and the formattedRect of the text as rectangle.

Runtime dependency

The implementation of pdf printing resides in a dynamic library called *revpdfprinter*. The standalone builder will automatically include this in the appropriate place when building standalones.

Any custom code that deals with standalones which use this feature must make sure that the *revpdfprinter* library resides next to the engine executable (On Mac, this is inside the Contents/MacOS folder, not next to the bundle).

Elevated process support (experimental)

Sometimes it is necessary to perform operations on the local machine as an administrator, and a typical pattern for a GUI application doing this is for it to prompt for authentication at certain points.

Modern operating systems do not permit a process to elevate itself, nor grant itself increased privilege. Instead, they only allow a running process to launch another process with increased privilege. Therefore, in order to support this, a new form of the **open process** command has been introduced that can launch a slave process with elevated permissions:

```
open elevated process process [ for [ text | binary ] ( read | write | update | neither ) ]
```

This form operates identically to the normal version, except that engine will ask the system to launch the given process with admin/root privileges.

The standard way for a GUI application that needs to perform privileged operations to be structured is to split the application into two parts: a GUI front-end that interacts with the user, and a command-line back-end that is run with elevated permissions. These two parts can then talk to each other using a standard master-slave approach, or some other form of IPC such as sockets.

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.

Slave process improvements

A number of issues with the *open process* command and the engine itself have, up until now, conspired to make it difficult (if not impossible!) to either run a slave process, or use the engine as slave on all platforms.

These issues have been resolved in this version, thus making it straightforward to run another process and poll for input and output over stdin/stdout.

The typical form for this is along the following lines (this example assumes the process being executed outputs whole lines):

```
command startSlave pProcess
  open process pProcess for text update
  send "monitorSlave pProcess" to me in 50 millisecs
end startSlave

command monitorSlave pProcess
  repeat forever
    # Loop until there are no more lines to read.
    read from process pProcess for 1 line in 0 millisecs
    if the result is empty then
      # The slave has sent us something, so process it and loop for
      # (potentially) more data.
    else if the result is "timeout" then
      # There is nothing waiting for us, so exit repeat
      exit repeat
    else if the result is "eof" then
      # The slave has terminated, so do any final processing and finish
      # monitoring.
      close process pProcess
      exit monitorSlave
    else
      # Some error has occurred!
      exit monitorSlave
    end if
  end repeat
```

```

    send "monitorSlave pProcess" to me in 50 millisecs
end monitorSlave

```

Videograbber improvements

Some work has been done on storing and fetching settings from the videograbber.

On Windows and Mac, audio and video settings are now correctly retrieved and set using **revVideoGrabSettings** and **revSetVideoGrabSettings**.

On Windows (when using DirectShow), the 'Camara Control' settings are now saved and restored via the grab settings commands. Specifically: pan, tilt, roll, zoom, exposure, iris, focus and flash.

Status icon support (experimental)

Windows, Linux and Mac all have an area where so-called 'status icons' can be displayed. On Windows this is the system tray on the bottom right of the start bar, on Linux this is typically the right of the panel at the top of the screen, and on Mac this is on the menubar.

The engine has support for adding a single status icon, and it can be configured using **the statusIcon**, **the statusIconTooltip** and **the statusIconMenu**:

```

set the statusIcon to imageId
set the statusIconMenu to iconMenuSpec
set the statusIconToolTip to toolTip

```

Here *imageId* is the id of the image you wish to use as the icon. It will be scaled down automatically to the appropriate size for the platform and then set. The *toolTip* specifies what message appears when the user hovers over the status icon.

The *iconMenuSpec* allows you to configure a menu that will appear when the user does a 'menu' click on the icon. This string uses a subset of the standard engine menu specification:

```
[ <tab> * ] [ '(' <label> [ '|' <tag> ]
```

Here the number of tabs determines the depth of the menu (i.e. use this to create sub-menus). The optional tag is used when calling the **statusIconMenuPick** message.

Before the engine displays the status icon menu, it will send a **statusIconMenuOpening** menu to the current card of the defaultStack. You can use this opportunity to change the icon menu before it is displayed, this is an analog to handling *mouseDown* in a menu button.

When the user selects an item from the dock menu, the engine will send an **statusIconMenuPick** message to the current card of the default stack:

```
iconMenuPick which
```

Here *which* will be a list of labels or tags (if specified) separated by '|' which determines which item was selected.

In addition, the engine will send the following message in response to clicks on the icon:

```

statusIconMenuClick button
statusIconMenuDoubleClick button

```

You can use these to perform an appropriate action.

Note: If you wish to display a menu from the status icon you must use the *statusIconMenu* property,

attempting to open a normal popup menu in response to one of the click messages is not guaranteed to work.

Note: This syntax is only implemented on Windows at the moment and replaces the previously unsupported use of **the icon** and **the iconMenu** for this purpose. The properties specified above will have no effect on Mac and Linux at this time.

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.

Listing sub-keys in the registry (experimental)

To get a list of sub-keys in the Windows registry use the following function:

```
listRegistry(parentKey)
```

This will return a return-delimited list of sub-keys, i.e. those keys which are direct children of the given *parentKey*. The specified key should be in the same format as the other registry functions.

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.

Cryptographic-quality random data (experimental)

A **randomBytes()** function has been added to enable generation of arbitrary amounts of cryptographic-quality random data. To use this feature, the syntax is:

```
randomBytes(byteCount)
```

The function will return *byteCount* random bytes and uses the OpenSSL library's random data generator to do so.

The random data is derived from non-predictable sources where possible, meaning that it can (for the most part) be considered truly random. This is contrast to the **random** function which uses a pseudo-random number generator and **randomSeed** value.

Note: Make sure the security library is included when building applications that use this function as a standalone. If the library cannot be found at runtime, the function will throw an error.

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.

SHA-1 digest support (experimental)

The SHA-1 digest of a block of data can now be computed by using the following form:

```
the sha1Digest of data
```

```
sha1Digest(data)
```

This function returns the sha-1 digest in the form of 20 (binary) bytes.

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.

it out as we need feedback to develop it further.

Public key encryption support (experimental)

The `encrypt` and `decrypt` commands now support RSA public key encryption. The new forms supporting this are:

```
encrypt message using rsa with ( public | private ) key key [ and passphrase passphrase ]
```

```
decrypt message using rsa with ( public | private ) key key [ and passphrase passphrase ]
```

Here *key* should be in PEM format, optionally protected by *passphrase*.

The maximum length of a message that can be encrypted using RSA is the size of the key in bytes – 11. So, for a 512-bit key pair, the maximum encryptable message size is 53 bytes.

Generating key pairs

Public-private key pairs can be generated using the **OpenSSL** suite of command-line tools. For example:

```
openssl genrsa -out private_key.pem 512
```

```
openssl rsa -pubout -in private_key.pem -out public_key.pem
```

Will generate a key pair of size 512-bits, placing the private key in *private_key.pem* and the public key in *public_key.pem*.

For more information on these utilities see <http://www.openssl.org/docs/apps/rsa.html> and <http://www.openssl.org/docs/apps/genrsa.html>.

Command variants

Use `encrypt` with a public key to encode a message that you only want to be decoded the holder of the private key.

Use `decrypt` with a private key to decode a message that a sender has encrypted with its corresponding public key.

Use `encrypt` with a private key to encode a message that a receiver can then verify has come from one of the holders of the private key (this is a *signing* operation).

Use `decrypt` with a public key to verify that a message has been encoded with the corresponding private key, and there has come from one of its holders (this is a *verify* operation).

Encrypting longer messages

For signing, the maximum length of an encryptable message isn't really an issue since typically in that scenario it will be some sort of hash that would be being encrypted.

For the more traditional encrypting scenario, however, the standard approach is to use public key cryptography to encrypt a random password which is then used with a symmetric cipher to actually encrypt the payload.

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.

HTTPS – automatic root certificate discovery (experimental)

In previous versions it was necessary to set the `sslCertificates` property to the root certificates that HTTPS connections should be verified against. Support has now been added to locate and load the root certificates installed (and kept up to date) as part of the OS.

This uses the standard root certificate keychain on Mac, the standard root certificate store on Windows and uses a number of heuristics to locate this information on Linux.

You can easily find out if the system-installed root certificates are being found by running the following command in the message box:

```
get url "https://www.google.com"
```

```
put the result & return & it
```

If this results in an error about verification failure then it is likely that root certificates have not been found. Please let us know (particularly on Linux) if you find this simple test fails, making sure you give us full details of your system (e.g. Linux distribution and version).

Note: Unfortunately this feature does not currently work correctly on Mac 10.6.x. For now, we advise including an appropriate root certificates collection with your application, as was previously necessary, and setting the `sslCertificates` property appropriately.

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.

Runtime execution stack configuration (4.5.1)

In order to be able to more reliably control the maximum level of recursion, a new global property `stackLimit` has been introduced.

This property allows a script to set (in bytes) the maximum size of the (runtime) stack the engine uses for recursive computation. A change in the setting will only take effect when all currently executing handlers complete, and at this time the stack size limit will be reconfigured to the given limit, or the nearest amount to it depending on available memory.

The `stackLimit` currently in effect can be fetched using **the effective `stackLimit`**.

The `recursionLimit` property is now bounded by the `stackLimit` – attempts to set the `recursionLimit` greater than the `stackLimit` will see it downwardly adjusted to the maximum current size allowed.

Note: The changes to the `recursionLimit` property and the new `stackLimit` property are only implemented on Windows at present.

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.

Keyboard layout syncing (4.5.1)

The `field` object will no longer attempt to sync the keyboard to the current text. This brings LiveCode (and applications built with it) inline with modern practice on both Windows and Mac

(the Linux engine has never tried to do this synching).

Standalone *allowInterrupts* default (4.5.1)

The default setting for the *allowInterrupts* property is now false in standalones. This change has been made to ensure that user's cannot inadvertently interrupt scripts in built applications.

MySQL driver improvements (4.5.1)

The MySQL client library has been updated to the latest version (6.0.0). In particular, this fixes a long standing issue with connection timeouts.

Additionally, SSL support now works correctly on all platforms (this requires the 'SSL & Encryption' support be included in the standalone settings when deploying).

XML Namespace support (4.5.1)

By default, revXML will process XML Namespace related tags and attributes in a way that is inaccessible to revXML. This can cause problems when a script needs to process the namespaces itself.

To resolve this issue two additional functions mimicking the behavior of revCreateXMLTree and revCreateXMLTreeFromFile have been added to revXML. These functions are:

```
revCreateXMLTreeWithNamespaces
revCreateXMLTreeFromFileWithNamespaces
```

The difference between these and the originals is that these two functions ignore namespace tags and attributes, and instead return them as part of the tree without doing any processing.

For example, consider the following XML tree:

```
<db>
  <bb:record>
    <bb:firstname>Bugs</bb:firstname>
    <bb:lastname>Bunny</bb:lastname>
    <zz:zip>10101</zz:zip>
  </bb:record>
  <xx:record>
    <xx:firstname>Bugs</xx:firstname>
    <xx:lastname>Bunny</xx:lastname>
    <xx:zip>10101</xx:zip>
  </xx:record>
</db>
```

Will get interpreted as follows:

revCreateXMLTree	revCreateXMLTreeWithNamespaces
db	db
record[1]	bb:record[1]
record[1]/firstname	bb:record[1]/bb:firstname
record[1]/lastname	bb:record[1]/bb:lastname
record[1]/zip	bb:record[1]/zz:zip
record[2]	xx:record[2]
record[2]/firstname	xx:record[2]/xx:firstname

```
record[2]/lastname
record[2]/zip
```

```
xx:record[2]/xx:lastname
xx:record[2]/xx:zip
```

The latter form is more appropriate in the case that an application needs the namespace information to perform correctly.

Note: This is breaking change from 4.5 which (erroneously) changed the behavior of `revCreateXMLTree` to be that of `revCreateXMLTree*WithNamespaces`.*

revBrowser improvements (4.5.1)

There is a long standing issue with `revBrowser` that causes browser instances to be lost whenever the stack it is attached to has its window re-created. Previously, cases where this would occur had to be avoided when a browser was present on a stack.

To resolve this problem a new property has been added to browser instances – `windowId`. The `windowId` property allows the stack to which a browser instance is attached to be changed after it has been created.

If the `windowId` is set to 0, the browser instance is temporary hidden. If the `windowId` is set to a valid stack `windowId`, the browser instance will move to that stack.

For example, to toggle the `resizable` property of a stack hosting a browser use the following code:

```
revBrowserSet pBrowserId, "windowId", 0
set the resizable of stack pBrowserStack to pNewResizableValue
revBrowserSet pBrowserId, "windowId", the windowId of stack pBrowserStack
```

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.

Out-of-bounds group scrolling (4.5.3 – experimental)

It is now possible to configure a group to turn off automatic clamping of scroll offsets when scrollbars are not present. To control this use the following properties:

```
set the unboundedHScroll of group to booleanValue
set the unboundedVScroll of group to booleanValue
```

When an unbounded scroll property is set to true and the corresponding scrollbar is not visible, it is possible to set the scroll for that axis to values outside that of the size of the content (both positive and negative).

The main motivation for this change is to allow much easier implementation of the 'bouncing' effect common in iOS controls. Indeed, after setting the relevant unbounded scroll properties, it is enough to just use the scroll value provided by a scroller object directly as the value passed to the relevant group scroll property.

If a scrollbar is made visible or unbounded scroll is turned off and the corresponding scroll property is outside of the content bounds, the group is scrolled to bring the value within the appropriate range.

Note: Prior to 4.5.3-rc-2, unbounded scrolling was always enabled when the corresponding

scrollbar was hidden. Adding these properties retains backwards-compatibility as the default is for unbounded scrolling to be off.

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.

Linux – font support improvements

The font support in the Linux engine has been completely overhauled.

The engine now uses the *pango* library for all text layout, and renders text using the *xf* library. Additionally, the **fontNames** and related global properties will now reflect the full collection of fonts installed on the system via the *fontconfig* mechanism (this is the font selection system that powers pango).

Legacy font support

If the linux distribution does not have the necessary libraries installed, the engine will fallback to using the old X11 font mechanism. This has vastly reduced support for unicode text, and renders without anti-aliasing.

You can force this fallback by passing `-xftoff` as a command-line parameter to the engine.

Linux – theme improvements

Some work has been done to improve the appearance of applications on Linux. Although still not perfect, the native theme support should now correctly handle themes that have a transparent/alpha blended background on some of their parts.

In particular, the default theme in the latest distributions such as Ubuntu 10 look much improved.

Linux – ssl improvements

In order to provide ssl and industrial strength encryption, the engine utilizes the *OpenSSL* library. However, the OpenSSL team does not, unfortunately, guarantee binary compatibility between different versions. This has long caused an issue for the engine on Linux where the specific version the engine needs may not be present.

In order to solve this issue, the linux engine no longer depends on any installed OpenSSL library and will, instead, (attempt to) load *revsecurity.so*.

This loadable library is a custom build of the latest OpenSSL distribution (1.0.0a at time of writing) and includes both the ssl and crypto components.

Runtime dependency

This change introduces a new (optional) runtime dependency for linux standalones. The standalone builder will automatically include this in the appropriate place when building standalones that have specified a need for the 'SSL and Security' library.

Any custom code that deals with standalones which use this feature must make sure that the *revsecurity* library resides next to the engine executable.

Mac – bundle changes

In previous engines there were various support bundles present in the *PlugIns* folder inside the engine bundle. These are no longer required as their functionality has been integrated into the main engine executable.

Any code that builds standalones, or manipulates built standalones will need to be updated to not expect these to be present.

Mac – dock icon support (experimental)

Previously unsupported syntax for manipulating the dock icon on Mac is now experimental.

Choosing an image

The current dock icon image can be set by using the global **icon** property:

set the icon to *imageId*

The engine will attempt to find an image with the given id, resize it to 128x128 and then set it as the dock icon for the application.

This property has no effect on other platforms.

Note: The image is only guaranteed to persist while the application runs, although in some cases the OS does appear to cache it beyond this.

Configuring the dock icon menu

In addition to changing the dock icon image, you can also configure the menu that appears when the user clicks on it.

To set the dock icon menu use the global **iconMenu** property:

set the iconMenu to *iconMenuSpec*

Here, *iconMenuSpec* is a string describing the menu. This uses a subset of the standard menu specification syntax. The string should be a return-delimited list of items specified as follows:

```
[ <tab> * ] [ '(' ] <label> [ '|' <tag> ]
```

Here the number of tabs determines the depth of the menu (i.e. use this to create sub-menus). The optional tag is used when calling the **iconMenuPick** message.

Before the engine displays the icon menu, it will send a **iconMenuOpening** menu to the current card of the defaultStack. You can use this opportunity to change the icon menu before it is displayed, this is an analog to handling *mouseDown* in a menu button.

When the user selects an item from the dock menu, the engine will send an **iconMenuPick** message to the current card of the default stack:

iconMenuPick *which*

Here *which* will be a list of labels or tags (if specified) separated by '|' which determines which item was selected.

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.

Mac – large progress bar (4.5.3)

The scrollbar object will now render a large progress bar when its height is greater or equal to 20 pixels and its style is *progress*.

Mac – window modification indicator (4.5.3 – experimental)

Standard Mac document windows typically display the modified state of the document they represent by adding a dark dot to the close-box. It is possible to control the display of this dot in LiveCode by using:

set the modifiedMark of *stack* to *booleanValue*

The indicator will be displayed when the property is set to *true*.

This property has no effect on Windows nor Linux.

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.

Windows – ssl improvements (4.5.1)

The SSL libraries required for encryption and HTTPS on Windows have been updated to use OpenSSL 1.0.0a and have been compiled into a single DLL – revsecurity.dll.

This DLL must be present in the same folder as the (standalone) engine for these features to work (note the standalone builder automatically includes this library if the appropriate inclusion is set).

Maintenance

Noteworthy changes

The following bug fixes are ones which may have an impact on existing code and/or may be important to a wide audience.

Intermittent heap corruption issue on Windows (4.5.3)

A heap corruption issue dating back to 4.5.1 has been discovered and fix. The issue will manifest itself as a seemingly random crash, and although quite rare has the chance to affect most LiveCode applications at some point. It is strongly recommended that all standalones built using 4.5.1 and 4.5.2 be updated to use 4.5.3 as soon as is possible.

UTF-8 Conversion does not take into account surrogate pairs (4.5.3)

Prior to 4.5.3, converting to or from UTF-8 using the *uniEncode* and *uniDecode* functions would not correctly process surrogates. In particular when converting from UTF-16 to UTF-8, the surrogate pairs would be encoded as two separate 3-byte characters rather than a single 4-byte character; when converting from UTF-8 to UTF-16, 4-byte characters would be ignored.

This has now been corrected. It is not expected that this change will have any noticeable impact on existing code, except making LiveCode more interoperable with applications that validate UTF-8

for wellformedness.

Any data in UTF-8 that contains pairs of 3-byte characters, rather than single 4-byte characters, will correct itself when it next roundtrips through uniEncode/uniDecode.

Bug 3604 – Maximize behavior uses incorrect windowBoundingRect on Mac (4.5.2)

The long standing issue of the windowBoundingRect on Mac being adjusted for MacOS Classic margins has been resolved. The full rect specified is now used to guide maximization on Mac.

Bug 7670 – handlers passing through groups do not pass through backgrounds (4.5.1)

Previously, if a handler passed through a group it would then skip and backgrounds on the card. This was contrary to documentation and to consistency of the message path. This issue has now been resolved, messages passing through groups will pass through the card, and then through any backgrounds just like any other control.

Bug 8563 – can't copy text from PDFs generated by LiveCode (4.5.3)

Prior to 4.5.3, PDF printing would not produce text output that could be reliably copied with PDF readers. This has been fixed in this version, both Unicode and non-Unicode text should be copiable (although how well will, in many cases depend on the text extraction capabilities of the PDF reader).

Bug 9180 – incorrect lang tag attached to html 'font' tags (4.5.1)

The lang tag will now only ever have the value 'en-UC', and only be present if the tag contains Unicode characters – this now correctly reflects its use inside the engine (to mark style runs that contain Unicode).

Bug 9193 – windows incorrectly placed on Linux (4.5.2)

The long standing issue of windows on Linux being placed relative to the top-left of the frame and not the client-area has been resolved. Code that attempts to adjust for the previously erroneous behavior will need to be removed to ensure windows are placed correctly on this platform.

Specific bug fixes (4.5.3)

(bug fixes specific to the latest build are highlighted in bold, bug fixes that have been reverted are stricken through)

Intermittant heap corruption bug on Windows

UTF-8 conversion not wellformed when surrogate pairs are involved

Strange results when using 'the result' and a function call in a condition

Speed regression caused by fix to 9218

The name of the IDE engine executable is now (correctly) LiveCode

Incorrect type of comparison used in specific (rare) cases on Windows

2817 Setting the icon of a stack to 0 causes the previously referenced image to disappear

8563 Can't copy text from PDF files generated by LiveCode

9204 Some stacks saved with 4.5 won't load into 3.5 linux standalone engine

9206 Setting the defaultCursor causes memory leak

9209 Frontmost stack can appear hilited even if app in background (refixed)

9210 Default button does not unhilite/stop pulsing when app in background

9211 No way to get 'large' progress bar on Mac

9218 Changing loop variable in 'repeat for each key' causes the array to be changed

- 9233 Window content disappears when you click twice on Maximize on Mac
- 9240 Flip doesn't reset angle to 0
- 9242 The screenRects works incorrectly on Linux with multiple monitors
- 9259 Crash on setting the defaultCursor in specific cases
- 9269 No way to set the 'window modified' mark on Mac
- 9277 Wrong httpHeaders can be used when executing load url

Specific bug fixes (4.5.2)

- hasMemory() now functions the same on Mac as the other platforms
- SQLite driver incorrectly processes integer columns (64-bit range now works correctly)
- 3604 Maximize behavior uses incorrect windowBoundingRect on Mac
- 3892 Ctrl+Alt not recognised as a synonym for AltGr on Windows
- 9175 Cursors drawn without alpha-channel in some cases on Windows
- 9175 Cursors of small sizes render corrupted on Windows in some cases
- 9180 Incorrect lang tag attached to htmlText 'font' tags
- 9182 Intermittant problem with replace command on some Linux distributions
- 9183 Failure to retrieve image data that SnagIt places on the clipboard
- 9190 Trying to build a revlet without appropriate deployment pack causes cryptic error
- 9193 Windows incorrectly placed on Linux
- 9194 selectedObjectChanged not sent when object deselected and 'exit to top' used
- 9195 'the angle' of an image has no immediate effect if the image has not been decoded

Specific bug fixes (4.5.1)

- 'No disk' exception on startup on some Windows machines
- PDF printing fails on Vista when rendering text
- Editing and selection of Unicode surrogates does not work correctly
- MySQL driver now returns appropriate error if SSL could not be established
- Language bar disappears when LiveCode has focus on Windows Vista/7
- Crash when relayering groups in response to mouseEnter
- Redraw issue when showing hidden stack in a specific case on Windows
- 4022 Immediate 'read from process' causes hang on Windows
- 5246 Put ... into url ("file:~...") causes crash when ~ does not resolve to a user
- 6044 System cursor theme ignored on Linux
- 7476 'edit script of ...' doesn't work if lock messages is in effect
- 7670 Handlers passing through groups do not pass through backgrounds
- 7864 Line length limit of 65535 chars in filter command
- 7892 RevBrowser vScroll/hScroll properties do not work on Windows
- 7988 SpecialFolderPath() can returns a non-empty result even if it succeeded on Windows
- 8107 Graphic object auto-closes polygons when it shouldn't
- 8151 Stack title changes when stack opacity goes from 0 to non-zero on Windows
- 8218 Delete/backspace doesn't work if the shift key is down
- 8259 'there is a folder' returns False for network shares
- 8352 allowInterrupts not false in standalones
- 8494 Clone fails on groups when in 'edit group' mode
- 8585 Single line tooltips on controls near the taskbar flash
- 8640 The hilitedLines doesn't allow indices > 65535
- 8847 Events and messages stop working in some circumstances on Linux
- 8972 RevSetVideoGrabSettings has no effect

- 8973 LiveCode crashes when fetching a 16-bit deep image from the clipboardData on Windows
- 8980 revDataFromQuery truncates output for NULL columns with SQLite
- 8994 Bucket and pencil cursors do not appear on Mac
- 8999 Landscape orientation does not work correctly with pdf printing
- 9005 Recursion limit can be set to an invalid value
- 9021 Cursor setting related crash on Mac
- 9023 The printRectangle is not updated after setting the print* properties
- 9027 Odd behavior of scrolling field after move command
- 9031 'print card into rect' requires explicit 'go to card'
- 9032 Crash when fetching revBrowser htmlText property
- 9034 Crash when doing 'filter empty ...' in certain cases
- 9035 Hierarchical submenus disappear if the first item is disabled
- 9041 Toggling resizable of a stack causes browser content to disappear
- 9048 Text in opaque fields does not render reliably on Linux
- 9055 Inconsistency of mouse messages in pointer tool mode
- 9059 capsLockKey function non-functional on Linux
- 9102 the printRanges reports invalid value, does not coalesce ranges correctly
- 9115 Non-Unicode text does not print to pdf correctly in some cases on Mac
- 9118 Fill patterns do not print correctly on Windows
- 9120 hostNameToAddress does not return immediately in some cases on Mac
- 9123 'clone invisible <object>' causes rendering artifact
- 9134 'openProcesses' not cleared if process quits on Windows
- 9149 LiveCode should not auto-sync keyboard to text
- 9178 'do as vbscript' fails after several runs (Windows)
- 9179 hostNameToAddress can cause LiveCode to crash (Windows)

Specific bug fixes (4.5)

- VideoGrabber settings can be reset to defaults before showing dialogs when using DirectX
- Volume control slider inaccessible in QT players on Windows
- Crash after setting the label of a graphic to empty in certain circumstances
- Engine fails to launch on Mac 10.3.9.
- crash when using 'open elevated process' on Mac
- 'open elevated process' fails on Linux if spaces in engine or command path
- 5 Images needs to be RLE compressed to be used as cursors
- 1413 Some PNGs do not import correctly
- 1690 'beep n' beeps (n-1) times
- 2117 Engine blocks on DNS query
- 2398 Cursor size limit
- 3568 Crash on using a non-RLE compressed/color image when setting a cursor
- 3977 Mac cursors are scaled down to 16x16
- 6215 Arrow keys give keyUp message after typing into field
- 6506 the watchedVariables doesn't support globals
- 6753 Rotated image moves when toggling dontDither
- 6866 Crash when fetching htmlText of field under certain circumstances
- 7490 Default buttons render incorrectly on Linux
- 7613 Crash when throwing an error in newButton handler
- 7649 Filter command incorrectly removing empty lines
- 7652 Some cursors turn white on Mac OS X

- 7666 revDatabaseCursors function adds trailing comma
- 7744 Parser allows duplicate parameters
- 7792 Handlers can be compiled with builtin function names even though they cannot be invoked
- 7815 Ask file can't handle folder name in path longer than 31 characters on Mac OS X
- 7872 Crash on testing if path containing ~<non-existent-user> exists
- 8064 Flip command crashes rev in certain circumstances
- 8143 Left and right quotes and double-quotes don't appear when pasted into Rev from WordPad
- 8264 Crash when using replaceText function in certain specific circumstances
- 8268 Crash when compacting stack after deleting cards with background group
- 8273 revDatabaseColumnIsNull reports False for NULL columns in SQLite
- 8280 Stack behavior not resolved on open
- 8327 Copying text from Safari can cause dataloss and strange characters on Mac OS X
- 8351 SQLite query errors returning unhelpful generic error messages
- 8427 Crash when exporting JPEG images in certain circumstances
- 8549 revBrowser not loading htmlText before a url has been opened on Windows.
- 8559 Setting the menuHistory of an option menu does not change the scroll position on Windows
- 8560 Incorrect hilighting of tab buttons after using menuHistory
- 8562 Crash when calling 'cancel printing' after 'open printing to pdf'
- 8564 Image borders don't print to PDF
- 8571 Setting the icon fails to set the dock icon on Mac OS X unless the image is edited first
- 8572 printMargins causes issues when printing to pdf
- 8573 Menu separators and disabled menu items not working in OS X Plugin.
- 8593 Scrollbar thumb on list field can disappear
- 8595 CMYK jpegs without an ICC profile fail to display
- 8597 Crash in some cases when using graphic effects on images
- 8598 Mac OS X tab buttons render with increased margin
- 8601 Cannot slide a slider if startValue > endValue
- 8610 Passkey property doesn't give any indication of whether the script is available
- 8620 revDatabaseTableNames truncating data with large number of tables
- 8621 revDatabaseTableNames doesn't remove trailing return character
- 8622 Buttons not working properly in licensing dialog
- 8624 Standalones not made executable on OS X if they had accented chars in path
- 8627 Spanish dates have no spaces on Mac OS X
- 8628 Problem when moving between different types of pulldown menu in specific circumstances
- 8650 Crash when operating ComboBox using keyboard on OS X
- 8661 show/hide with visual effect doesn't take into bitmap effects
- 8662 Printing to pdf doesn't take account of the printPaperOrientation
- 8671 SQLite incorrectly returning empty for columns in some queries
- 8716 Stacks containing large custom prop arrays won't load
- 8746 Resizing a menubar group, or a menu button causes its accelerators to stop working.
- 8769 Crash when using graphic effects on Linux in some cases
- 8773 Player object cannot use https streams
- 8848 Context menu clicks on datagrids in the IDE causes contents of the datagrid to be lost
- 8888 In list behavior mode, 'the hilitedLines' are not preserved across fetching htmlText chunks
- 8889 Video grabber set settings crashes if given an invalid parameter
- 8929 Shell command blocks when it shouldn't
- 8936 HTTPS only works if sslCertificates has been set correctly
- 8943 Blocking url call doesn't start until an event message has been sent
- 8946 Crash in script editor when editing specific script

- 8947 Drag cursor doesn't get set to 'not allowed' on Mac OS X
- 8956 QT Unloaded when it shouldn't be causing delay in first visual effect.
- 8968 Manipulating the currentCard of the templateStack causes a crash
- 8969 shell() does not block when it should on Windows

IDE changes

Getting folder locations within the IDE

If you write plugins, or have code that relies on the location of IDE files then please ensure you use the following access functions to locate them:

<code>revEnvironmentToolsPath()</code>	The location containing the main IDE files.
<code>revEnvironmentToolsetPath()</code>	The location of the main IDE stacks.
<code>revEnvironmentExternalsPath()</code>	The location of the externals that come with the IDE.
<code>revEnvironmentPluginsPath()</code>	The location of the plugins that come with the IDE.
<code>revEnvironmentRuntimePath()</code>	The location of the standalones that come with the IDE.
<code>revEnvironmentDocumentationPath()</code>	The location of the documentation files.
<code>revEnvironmentResourcesPath()</code>	The location of the resources that come with the IDE.
<code>revEnvironmentCustomizationPath()</code>	The location of the IDE customization folder.
<code>revEnvironmentUserCachePath()</code>	The location of the folder to use for caching files.
<code>revEnvironmentUserPreferencesPath()</code>	The location of the folder to use for preference files.
<code>revEnvironmentUserExternalsPath()</code>	The location of the folder to use for additional externals.
<code>revEnvironmentUserPluginsPath()</code>	The location of the folder to use for additional plugins.
<code>revEnvironmentUserResourcesPath()</code>	The location of the folder to use for additional resources.

Important: Third-party IDE extensions must avoid placing any files inside the application bundle or under `revEnvironmentToolsPath()` (not least because you will probably not have privileges to do so!). Instead, they should use the user-externals and user-plugins paths as provided. These paths are determined by the user's customization path setting, configurable in the preferences.

Update checker

The update checker and its corresponding menu item in *Help* have been temporarily removed. The version checking method is changing, and this option will re-appear in a subsequent release.

Affiliate registration

The affiliate registration item in the *Help* menu has been temporarily removed. The method of affiliate registration is changing, and this option will re-appear in a subsequent release.

Database query builder

Use of the database query builder has been deprecated.

To access its functionality, choose the *Enable database query builder* option on the *Compatibility & Updates* pane of the preferences.

Note: *The database query builder will be removed entirely in the next or subsequent non-maintenance release.*

iOS Support (4.5.2)

The IDE now contains built-in support for simulating and deploying iOS applications. The following features have been added to support this:

- An iOS pane has been added to the Standalone Builder
- *Save as Standalone Application* can now build iOS applications
- A *Simulate* button has been added to the menubar to enable fast and easy access to running applications in the simulator
- Configuration of simulator version and type can be accessed via a *Simulator Version* submenu in the *Development* menu.
- A *Mobile Support* pane has been added to the preferences to enable configuration of iOS SDKs (needed by deployment and simulation).

More details on how to use these features can be found in the *iOS Release Notes* accessible via the *Help* menu.

Note: If you do not have an iOS deployment pack, you can still use the iOS feature in trial mode. In this case the applications you build will contain a forced banner on startup lasting 5 seconds, and they will quit after one minute.

Access to resources folder (4.5.2)

The IDE comes with a Resources folder containing the sample projects, example stacks and other miscellany.

Previously this was easily accessible on all platforms by navigation to the installed IDE folder. As this folder is now (on some platforms) inside an application bundle, a new item in the Help menu *Example Stacks and Resources* has been added to open up the folder in the file manager on the running system.

Standalone builder

MacOS Classic

Support for MacOS classic building has been removed. If you wish to build MacOS classic applications then you must use a previous version to do so.

Windows – U3 Support

Support for building for U3 on Windows has been removed. If you wish to build applications for U3 then you must use a previous version to do so.

Windows – UAC Manifest

It is now possible to specify what action UAC should take on Windows Vista and higher when the standalone is launched. You can choose one of the following options:

- | | |
|------------------------|---|
| <i>Default</i> | No UAC option is provided in the manifest. |
| <i>Save as Invoker</i> | The application will run with the same privileges as the process that |

invoked it.

Highest available The application will be elevated to the highest privilege level the current user is allowed.

Require administrator The application will be run as administrator after prompting the user for appropriate login/elevation rights.

Web

Using the 4.5 standalone builder to build for Web will create revlets. These are compatible with the existing revWeb plug-in available from revweb.runrev.com.

An updated version of the IDE for producing LiveCode Applets and the associated LiveCode player will be made available in due course.

Note: As the currently available revWeb plug-in uses the 4.0 engine, you must be careful to only use features that are present in that version.

iOS (4.5.2)

There is now a pane for configuring iOS applications in the Standalone Builder. At this time, iOS standalone building has the following limitations (compared to desktop/web deployment);

- A stack configured for iOS cannot also deploy to other platforms
- The *Copy Referenced Files* feature is not implemented for iOS builds.
- The *Inclusions* and *Property Profiles* feature is not implemented for iOS builds.
- The *Stacks* configuration options are not available for iOS builds.

More details on how to use the iOS settings pane can be found in the 'iOS Release Notes', accessible via the Help menu.

Datagrid

The datagrid has been updated to version 1.0.2 build 11. The following changes have been made to it since 1.0.1:

1.0.2 build 11 dgDragMove was not setting the dragAction to "move"

Added dgColumnLabelEncoding[pColumn] property. Returns "mac" or "iso" depending on the platform the dgColumnLabel[pColumn] was set on. Only returns a value if label is set with a version of the behavior >= 1.0.2 build 11.

Setting the dgColumnLabel now stores an encoding property of "iso" or "mac" depending on the platform the label is set on.

Setting the dgLabel property of a column header now includes the encoding based on the platform the label was set on: dgLabel [pEncoding]. This applies to the Default Header behavior.

1.0.2 build 10 ColumnControlOfIndex now returns the long id of the control. The id being returned was only to be used internally and would not work outside of a script running in the data gid group itself.

- 1.0.2 build 9 RefreshList now maintains the hilited line (index) in the data grid.
- 1.0.2 build 8 Removed unnecessary lock/unlock screen in dgMouseDown when clicking with left mouse button.
You can now pass multiple key/value pairs to SetDataOfIndex/Line.
RefreshList now works with cached controls.
_CreateDragImageFromIndex and _CreateDragImageFromControl now reset the templateImage.
Fetching dgText and dgData now restore persistent data if the control has yet to be opened.
- 1.0.2 build 7 "column labels" was returning a comma delimited list instead of a line delimited list.
arrowKey message is always passed now. Previously it was not passed if the data grid acted on the arrowKey message.
The internal function that returned the list of visible controls was not always returning the proper controls and/or order of controls for tables.
When refreshing data in indexes when the data grid had focus the focused control would revert to the card rather than staying within the data grid.
Message catcher in dgList behavior was watching for setProp dgDataControl instead of getProp dgDataOfControl.
Clicking in the vertical scrollbar area no longer selects rows when the scrollbar isn't displaying a thumb.
Added dgClickLine, dgClickIndex and dgClickedInDataView.
You can now set "dimmed hilite color".
Added getprops for "text color", "effective text color", "header text color" and "effective header text color".
_table.RepositionColumns was not setting "lock messages" to value it had when entering the handler.
Added in getProp for "border color" and "effective border color".
- 1.0.2 build 6 AddLine now returns the line number of the newly added data.
Added ColumnControlOfLine.
Registry is no longer accessed unless registryRead is among the items of the securityPermissions property.
Column header label, horiztonal and vertical scrollbars no longer have traversalOn set to true.
The Data Grid no longer deletes row/column controls when setting the dgData/dgText or deleting the last row in a data grid. This means no more error messages if any of the above events occur as the result of a handler called within a row/column behavior script.

A `ResetData` message is now sent to rows (form) or column (tables) instances of the templates whenever it is being hidden and the data being displayed is empty. This allows the developer to reset any controls and clear any caches.

Instances of a data grid form row/table column templates now receive the `PreFillInData` message when data in the row is being cleared and new data inserted. This allows the developer to clear any caches if need be.

Messages sent to rows/columns are now trapped before getting to the Data Grid group. This insulates the outside world from these internal messages if not handled in the template behavior script.

- 1.0.2 build 5 `dgColumn` name no longer accepts an empty string for a column name.

- 1.0.2 build 4 Internal private function that returned list of visible controls would return more controls than there were actual records for tables that had less records than visible rows. This caused issues with visual feedback during drag reordering.

When drag reordering the drop indicator was not being positioned properly when dragging above the first line of the grid.

Drop indicator now draws all the way to the right in Data Grid forms.

- 1.0.2 build 3 Custom column controls now inherit the standard column behavior or the "default column behavior" (if set) if no behavior has been assigned to the control.

- 1.0.2 build 2 Column headers were not properly aligned when redrawing the Data Grid.

Column header labels were not being properly positioned when rednering if header alignment was centered.

- 1.0.2 build 1 If a Data Grid table didn't have any data in it then the horizontal scrollbar would not keep the column headers and columns in sync if the column width was wider than the visible area of the Data Grid.

Setting "fixed row height" now calls `ResetList` internally. This ensures that all data and accompanying row hilites are redrawn properly based on new setting.

`_VisibleSequences` was not returning proper values in all cases. This affected `dgVisibleLines`.

Added "header divider color" and "header divider threed color"

`AddData` could potentially send "FillInData" with messages locked if a developer had locked messages before calling it.

Maintenance

Noteworthy changes

The following bug fixes are ones which may have an impact on existing code and/or may be important to a wide audience.

Bug 9214 – Default User-Agent string changed in libUrl (4.5.3)

The default User-Agent string has been changed in libUrl from 'Revolution' to 'LiveCode'. Any server-side checking code for the origin of requests that uses 'Revolution' should be updated to also check for 'LiveCode'.

Note that you can still change the default by use of *the httpHeaders* property, or the *libUrlSetCustomHTTPHeaders* command.

Bug 9267 – revLicenseType returns a non-sensical value (4.5.3)

In previous versions of 4.5.x, revLicenseType() returned the license's user count rather than the edition type. In 4.5.3 and later it returns 'Enterprise' as this is the closest old-style edition to LiveCode which no longer has a concept of edition. (This change has been made to help with backwards-compatibility, however please note that this function is deprecated and will be removed entirely in a future release).

Specific bug fixes (4.5.3)

(bug fixes specific to the current build are highlighted in bold)

- LiveCode document icon does not appear for stackfiles on Mac
- MinimumOSVersion not written in iOS standalone plist properly
- Simulate choices not saved in preferences
- Certain stack names confuse 'Simulate' and other IDE functions
- Incorrect provisioning profile used for deployment when more than 9 are installed
- Simulate menu item not disabled when it should be
- 'Import snapshot from selected object' does not work

- 9214 User-Agent string not updated in libUrl
- 9217 Trying to quit when the debugger is active hangs LiveCode on Mac
- 9229 Functions with no spaces in the parameters do not work in the message box
- 9231 Legacy driver setting not saved/updated in preferences
- 9238 Simulate doesn't warn about invalid folder names in bundle
- 9241 Calls to non-existent stack in revPreferencesGUI
- 9253 Lists scroll out-of-bounds in revOnline
- 9267 revLicenseType returns a nonsensical value

Specific bug fixes (4.5.2)

- Simulate button does not hide when menubar icons turned off
- Standalone Builder does not detect use of 'randomBytes' when checking for inclusions

Specific bug fixes (4.5.1)

- Whitespace not stripped from fields in activation dialog
- 'File > Close and Remove From Memory' non-functional
- 7411 Nudging objects with arrow keys does not work when Suppress Messages is on
- 7995 RevMail fails with messages > 2000 chars on Windows
- 8138 Image library throws an error then '(This stack' item selected
- 8600 IDE does not work correctly with Suppress Messages is on
- 8655 Content-Length sent by libUrl causes issues with some servers
- 8940 IDE does not take 'cantSelect' property into account when testing for edit mode
- 8982 Spelling error in revEnvironmentExternalsPath in release notes

- 8983 Typo in get url example in release notes
- 8988 Script handler list not updated when it should be in script editor
- 8990 Printer Chooser throws an error in preOpenCard
- 8995 Accented characters not displayed correctly in the splash screen
- 9016 Printer dialog 'printer list' has disabled scrollbar when it shouldn't
- 9017 Printer dialog misses 'current' print range option
- 9024 Printer dialog does not set some print* properties
- 9026 Script error when turning Windows builds on or off in Standalone Builder.
- 9069 Standalone builder requires stack to have '.rev' or '.livecode' extension
- 9070 Duplicate object non-functional
- 9090 Go commands disabled in View menu
- 9158 New Control menu item disabled when it should be enabled
- 9160 Missing shortcuts from menubar
- 9165 Group context menu item 'Lock Location' doesn't work.

Specific bug fixes (4.5)

- Standalone builder creates incorrect plist for Mac and fails to include default icon.
- Corrected 'LiveTalk' reference in about dialog.
- Resource Center can open in a corrupted state.
- Revolution appears in the re-license dialog.
- revPrintField does not work well with 'open printing to pdf'
- Standalone Builder doesn't include externals
- Standalone Builder doesn't include revsecurity.so when needed on Linux
- Changed references to Revolution to LiveCode in various dialogs and windows
- Standalone settings does not update 'PDF Printer' check box correctly
- User image libraries not sorted in drop-down lists
- User image libraries created with conflicting ID ranges
- 5117 Custom functions called from msg box can't use variable values from debugger
- 6505 WatchedVariables not supporting globals.
- 7031 Icon chooser does not display user image libraries
- 7136 lockMessages being displayed in the Message Box list makes no sense
- 7291 Property inspector windowShape field behavior not the most friendly
- 7700 Property inspector doesn't allow firstIndent to be set back to zero
- 7879 Standalone builder won't allow custom plist to be removed from settings
- 7929 revNumberOfRecords function incorrectly documented return value
- 7970 Standalone builder gives unhelpful warning when attempting to include Oracle driver in Linux build
- 7992 Data Grid cannot be created from main IDE menu bar
- 8002 Property inspector doesn't allow changing stack name where only case is different
- 8023 Inconsistency between creating Data Grid by dragging and by double clicking
- 8046 Column resizing button in Data Grid inspector not disabling correctly
- 8079 Dictionary doesn't reset scroll when changing entry
- 8088 Script editor resizing problem on Mac OS X
- 8142 Polygon tool can't draw polygons with more than 10 sides
- 8147 Duplicate stack error when resuming suspended development tools
- 8148 Various IDE windows resizing problems
- 8152 Recent stack icons in Start Center not responding to mouse click
- 8157 Stacks with "!" at beginning of name don't show properly in IDE Window menu

- 8179 Launch document documentation needed improvement
- 8206 Text clipping problem in Property Inspector
- 8236 Logging message left in IDE
- 8286 Message box not showing error hints
- 8291 Command + A doesn't work to select all in script editor find field
- 8299 Command+w closes debugger when variable visualizer is focused
- 8300 Documentation incorrect for password property
- 8311 Documentation example problem for preOpenControl message
- 8321 Script editor documentation pane doesn't disambiguate clashing keywords
- 8376 Cancelling revBrowser navigation using browserCancel doesn't work
- 8388 Script editor handler list not updating when closing tab
- 8400 Build for Mac OS Classic checkbox incorrectly enabled on Windows and Linux
- 8431 Typo in message box scrolling code throws error
- 8438 Installer is PowerPC only and requires Rosetta to run
- 8450 Messages sent to stack while building standalone with externals
- 8454 revStudio dock icon badging is wrong
- 8470 Standalone build fails when building for multiple platforms and using cursors
- 8486 Dictionary resizing error
- 8491 Documentation entries for revTalk objects contained incorrect syntax section
- 8497 Variable viewer panes being incorrectly hidden when debugger regains focus
- 8523 Line number in Script Editor overlaps horizontal scrollbar
- 8536 Database external not included in Linux standalone with search for inclusions
- 8546 Message box too picky about function syntax
- 8548 Script editor handler list not updating after unlocking passworded script
- 8579 Menu builder preview bar not working properly due to extra controls
- 8608 Stacks with trailing spaces in their names fail to build on Windows
- 8651 Graphic with showName and lineSize of zero breaks property inspector
- 8655 LibUrl sending incorrect content-length header
- 8715 Command-A not working on dictionary
- 8719 Mnemonics for View and Edit in the main IDE menubar are the same
- 8734 Application browser not working well with edit group mode
- 8744 Images emptied by standalone builder with Copy Referenced files turned on
- 8965 Pasting text into the S/E find field causes it to be overwritten rather than inserted
- 8966 Cancelling installation does not work correctly.
- 8970 Ctrl-C fails in variable visualizer

Revisions

- | | | |
|-------------------|----|---|
| <i>Revision 1</i> | MW | Document created for issue with 4.5-dp-4. |
| <i>Revision 2</i> | MW | Updated engine bug fix list for 4.5-dp-5.
Updated IDE bug fix list for 4.5-dp-5.
Added section on new public-key cryptography support.
Added section on new randomBytes support.
Added section on SHA-1 digest support.
Added section on root certificate discovery support. |
| <i>Revision 3</i> | MW | Updated engine bug fix list for 4.5-rc-1.
Updated IDE bug fix list for 4.5-rc-1.
Revised section on new licensing system |
| <i>Revision 4</i> | MW | Fixed broken engine bug fix list. |

<i>Revision 5</i>	MW	Updated engine bug fix list for 4.5-rc-2. Updated IDE bug fix list for 4.5-rc-2. Updated standalone builder 'Web' section.
<i>Revision 6</i>	MW	Updated engine bug fix list for 4.5 release. Updated IDE bug fix list for 4.5 release. Updated licensing section.
<i>Revision 7</i>	MW	Updated engine bug fix list for 4.5.1-dp-1. Updated IDE bug fix list for 4.5.1-dp-1. Updated datagrid section for 4.5.1-dp-1. Updated 'Improved cursor support' section Corrected example in 'slave process' section Added section on changes to recursion and stack handling Added section on improvements to MySQL driver Added section on improvements to revXML Added section on changes to SSL support on Windows Added section on improvements to revBrowser
<i>Revision 8</i>	MW	Updated engine bug fix list for 4.5.1-dp-2. Update IDE bug fix list for 4.5.1-dp-2.
<i>Revision 9</i>	MW	Updated engine bug fix list for 4.5.1-dp-3. Updated IDE bug fix list for 4.5.1-dp-3. Added section on allowInterrupts in standalones Added section on keyboard synching in fields
<i>Revision 10</i>	MW	Updated engine bug fix list for 4.5.1-rc-1 Updated IDE bug fix list for 4.5.1-rc-1
<i>Revision 11</i>	MW	Updated engine bug fix list for 4.5.1 release Updated IDE bug fix list for 4.5.1 release Updated revEnvironment function section.
<i>Revision 12</i>	MW	Updated engine bug fix list for 4.5.2-rc-1 release Updated ide bug fix list for 4.5.2-rc-1 release Added section on iOS Support Added section on iOS deployment to Standalone Builder section Added section on access to resources folder Added note on inclusion requirement to randomBytes section
<i>Revision 13</i>	MW	Updated and reorganized engine bug fix list for 4.5.2 release Updated and reorganized ide bug fix list for 4.5.2 release
<i>Revision 14</i>	MW	Updated engine bug fix list for 4.5.3-dp-1 Updated ide bug fix list for 4.5.3-dp-1 Added section on Mac large progress bar support Added section on group out-of-bounds scrolling Changed references to Mac OS X to Mac Updated randomBytes section to mention throwing of errors
<i>Revision 15</i>	MW	Updated engine bug fix list for 4.5.3-rc-1 Updated ide bug fix list for 4.5.3-rc-1 Reworked initial sections changing into 'Setup' Added section on multi-user setup support Added a note to automatic certificate discovery about 10.6.x
<i>Revision 16</i>	MW	Updated engine bug fix list for 4.5.3-rc-2 Updated ide bug fix list for 4.5.3-rc-2 Added information about unbounded scroll properties to <i>out-of-bounds</i>

scrolling section.

Added a 'Noteworthy Changes' section to IDE Maintenance

Added the change to user-agent string to IDE noteworthy changes

Added heap corruption bug to Engine noteworthy changes

Revision 17 MW Updated engine bug fix list for 4.5.3-rc-3

Updated ide bug fix list for 4.5.3-rc-3

Corrected set example in *out-of-bounds scrolling* section.

Revision 18 MW Updated engine bug fix list for 4.5.3-rc-4

Updated ide bug fix list for 4.5.3-rc-4

Added section on modifiedMark property

Revision 19 MW Updated engine bug fix list for 4.5.3 release

Updated ide bug fix list for 4.5.3 release