

Eggplant for Linux Release Notes

The notes below provide descriptions of the new features and changes introduced with each release of eggPlant. You are strongly encouraged to read about the relevant changes whenever you upgrade from an earlier version.

Potential compatibility issues are highlighted in **RED**.

Notes For Cross Platform Users:

Eggplant for Linux contains almost all of the functionality available in Eggplant for Mac. Excluded features are listed at the bottom of the release notes.

Release v11.10

(16-November-2011)

Highlights:

- Added an interface for managing licenses directly within the application. (Internet Connection Required)
- Added an option for eggPlant to check for updates. (Internet Connection Required)
- Added support for RealVNC Mobile connections on Windows (Android, Blackberry, Symbian S60 and Windows Mobile)
- **This will be the final release with support for Ubuntu 8 and earlier.**
- **This will be the final release with support for Fedora 12 and earlier.**

User Interface:

- Added a setting to change (or disable) the Countdown Delay when the Manual Image Doctor engages.
- Script names in the Results pane of the Suite window are now shown in green or red to reflect whether the most recent run of that script resulted in success or failure.

Scripting:

- Improved ReadText() and ReadTable() to return multi-line strings separated by **return**.
- Added the ability to find OCR text that includes **return** in the string.
- Added CaptureDate and CaptureHost to the information recorded when an image is captured, and available through the ImageInfo() function.
- Enhanced the ability to identify strings when finding OCR text.
- You can now specify a HotSpot override when using OCR text finding commands.
- Added the ability to specify a -username when making a connection via the CLI.

Bug Fixes / Tweaks:

- Fixed problem with launching Eggplant for Linux on multiprocessor machines.

- Added the ability to drag and drop text within a Script Editor window (when this feature is enabled in the Preferences).
 - Fixed the shell() command on Windows to handle path's with spaces in the path name.
 - Restored crash log submission on Mac OS X 10.7 (Lion).
 - Dramatically reduced memory use when doing long waitFor's or imageFounds within a script.
 - Fixed a number of memory leaks doing intensive repeat loops.
 - Fixed an occasional hang on Mac when a connected rendezvous server goes away.
 - Fixed problem that OCR property settings passed as an options parameter to ReadText or ReadTable would persist for future calls (stored in the readTextSettings global property).
 - Fixed a problem on Mac OS X 10.7 (Lion) where you couldn't paste into the find/replace panel.
 - Fixed an exception when changing the **predefinedVariables** on Mac OS X 10.7 (Lion).
 - Prevent creating a new Suite in place of an existing suite.
 - Prevent **Save As...** for Suites which would copy all data.
 - Improved the behavior in the Save Image panel when attempting to create a new folder with the same name as an existing folder.
 - Fixed some problems in the Save Image panel when a subfolder has the same name as one of its parent or ancestor folders.
 - Fixed problem with the creation date of the Eggplant.app itself.
-

Release v11.03

(09-October-2011)

Highlights

- Added embedded required libraries for OCR engine for compatibility on systems where libraries had been updated.

Bug Fixes / Tweaks:

- Properly restore connection size after a connection is closed abnormally.
 - Fixed a problem where Eggplant would allow scripts to fail when a connection was reconnecting.
 - Fixed a problem where Eggplant would report no active connection when a connection was reconnecting.
 - Fixed a lockup problem when Eggplant is reconnecting while screen is being searched.
 - Fixed the inability to using arrow keys in certain text fields within the Suite Window.
 - Fixed a crashing problem when an image in a sub-folder was selected when another image updated.
 - Fixed a rare problem where images would not show up in the Images Tab after being added.
 - Removed temporary TEXT properties, with a *, when using FoundImageInfo().
 - Fixed a crash following an exit repeat during iteration of a list, if the list is then iterated fully.
 - Removed a spurious warning about the "Spell Checker" when running an ask or answer command.
 - Support Submit panel no longer submits when return is pressed.
 - Fixed a problem when launching multiple instances of Eggplant simultaneously, reporting GSAApplicationLock already locked.
-

Release v11.02

(12-September-2011)

Bug Fixes / Tweaks:

- [Added support for RedHat Enterprise 5 Distributions of Linux.](#)
- Pastes from the main menu will now remove formatting characters and colors.

- Made the ReadText and ReadTable commands conform to the RemoteWorkInterval delay before reading values from the screen.
 - Fixed a problem with certain newline characters showing up in the logs.
-

Release v11.01 (25-August-2011)

Bug Fixes / Tweaks:

- **This release is NOT compatible with Red Hat Enterprise 5.x. A subsequent releases will restore RHE 5.x support.**
 - Added the ability to specify image names for the SearchRectangle of an OCR mouse command.
 - Fixed a problem with ReadText() when the searchRectangle was set.
 - Fixed a problem with the Script Editor context (eg: right-click) menu.
 - Fixed a potential hang when connections needed to be re-established.
 - Added greater details when unable to verify a valid Floating License.
 - Fixed a problem where the release notes could reopen every time the application launched.
 - Added support to run on SELinux platforms without needing "Permissive" set.
 - Added support for UltraVNC 1.09 connections.
 - Fixed a crash after setting the searchRectangle.
-

Release v11.0 (15-August-2011)

Highlights

- Added a multi-purpose **ReadText()** function to read text from any screen using OCR.
 - Added the ability to search all scripts in a suite.
 - Added find and replace capabilities to script editor.
 - Added the ability to easily repeat part of a script for a given length of time, or until a specified time.
 - Added support for **RealVNC Mobile 1.5** Connections.
 - There is now a generic Text Search engine for finding text strings on the screen.
If you have NOT modified your text preferences this is now the default Text Platform.
 - Fixed a number of bugs and compatibility issues – (See Below).
 - **This release is currently NOT compatible with Red Hat Enterprise 5.x. A subsequent releases will restore RHE 5.x support.**
-

User Interface:

- Added a Search Field for Scripts, allowing you to search script names or the full content of every script in the suite.
- Added a "Last Run" column on the Scripts pane of the Suite window, that shows the time when each script was last run. The time for a successful run will be shown in green; for a failed run in red. Clicking the last run time will show the results for that script run in the Results pane.
- Added a connection type pop-up to the VNC Connection panel that allows specifying explicitly which protocol to use for a VNC Connection.
For users with mobile enabled this includes the RealVNC Mobile Connection type.

- Enabled editing of script names directly in the Scripts tab. Simply select a script then click its name a second time to begin editing. Double-clicking quickly will still open the script.
- Added support for find and replace in the find panel when editing scripts.
- Eggplant will no longer allow you to create two images by the same name in the same folder, regardless of what file type extensions they use.

Scripting:

- Changed the initial setting of **the defaultLineDelimiter** to be the set of all common line endings (CRLF, LF, CR, LineSeparator, ParagraphSeparator) rather than just the Return (LF) character.
- Added **ReadText()** function which takes one or two parameters. The first (required) parameter is either a rectangle specifying the area of the remote screen from which text is to be read, or a point or image indicating the location of some text to read. The function returns whatever text was recognized from that part of the screen. In the case of a rectangle, it will read all text contained within that rectangle. In the case of a point, it will attempt to locate text near that point, and will read and return as much text as it can find adjacent to that text on the same line.

The second (optional) parameter is a property list of TextReading options.

Example:

```
put readText((40,0,400,30))
--> Finder File Edit View Go

put readText((200,10))
--> Edit
```

- Added a **HighlightRectangle** command which draws a rectangle highlight in the Remote Screen window. It requires a first parameter which is the rectangle to highlight. An optional second parameter is a property list of options, which may include color and duration properties. The **color** option specifies the color to use for drawing the rectangle, which defaults to red. The **duration** property specifies the length of time the highlight rectangle will be shown. If not specified it will be shown for the default ImageHighlightDuration time. Specifying zero for the duration will cause the highlight to remain until the next time a highlight rectangle is shown, either by another call to this command, or by any screen search command that draws highlights.
- A new Text Finding Engine (**OCR Search**) now exists in the Eggplant Preferences -> Text section. **If you have NOT modified your text preferences this is NOW the default Text Finding Engine.** If you have modified your text preferences you can create a new platform that uses OCR. You can always change your Default Text Platform using the Text Preferences. When using the **OCR Search** text engine you can specify text in any image finding command. You don't need to specify font characteristics but you can select language options for various styles.

Example:

```
click(Text: "File") // If OCR is your default platform
moveTo(Text: "1:34 AM", TextPlatform:"OCR") // If OCR is not the default platform
```

- The **RunWithNewResults** command will no longer change the status of any connections. Connections that are open when the command is issued will stay open when the called script begins, and connections that are open when the called script finishes will stay open in the calling script.
- Added an **every** option to the **offset** and **range** functions, to obtain the offset location of every occurrence of one text string within another or the full range (starting and ending positions) of every instance of some text within a larger text string. In addition to the natural language syntax, these can

also be called using new **everyOffset()** and **everyRange()** functions.

```
put every range of "at" in "concatenation" into atRanges
put atRanges -- (5 to 6,9 to 10)
```

- Added initial support for **units** associated with numeric values. Any variable holding a numeric value can have a unit type associated with it. Two types of units are officially supported at present: a time interval (or duration) type with a unit of **seconds** and a byte size (or data size) type with a unit of **bytes**. Values created with the special syntaxes for these types will automatically be assigned with those unit types:

```
put 2 hours into waitTime -- a duration with units of 'seconds'
put 16 megabytes into capacity -- a byteSize with units of 'bytes'
```

- Added the **unitsEnabled** global property to specify whether units should be shown when displaying a value (the initial setting is false):

```
put 1 minute 30 seconds into delay
put delay -- 90
set the unitsEnabled to true
put delay -- 90 seconds
```

- Added the ability to **repeat for duration** to create a repeat loop that terminates after a specified length of time has passed. This type of repeat loop will execute the body of the repeat block, then check whether the specified length of time has elapsed since the repeat began. If so, the loop is terminated, otherwise the body of the repeat loop is executed repeatedly with the time being checked at the top of the loop each time until the specified time duration has passed.

```
repeat for 3 seconds
  add 1 to counter
end repeat
put "Counted to " & counter & " in 3 seconds"
```

- Extended the **is a** operator to allow testing whether a value is a particular type of unit. For example you can test whether a value is a duration using the "is a duration" (or "is a timeInterval") operator, or whether it is a byte size using the "is a byteSize" (or "is a dataSize") operator:

```
put 3 minutes into x
put x is a duration -- true
put 5 megabytes is a dataSize -- true
```

- Added the **units** property of a variable to access or change the units associated with that variable. For timeInterval values, the units will be "seconds"; for a byte size value it will be "bytes".

```
put 1 minute 30 seconds into delay
put delay's units -- seconds
set delay's units to "bytes"
```

Other units may be used but are not specifically supported, so should be used experimentally with the understanding that the behavior may change as new unit types are added in the future:

```
set the unitsEnabled to true
put 12 into snack -- (must be a number before assigning units)
set snack's units to "peanuts"
put snack -- 12 peanuts
subtract 5 from snack -- (eat a few)
put snack -- 7 peanuts
```

- Added ability to **repeat until** *time* to allow a script to repeat a series of steps until a particular date/time has arrived.

```
repeat until "5:00 PM"
    doSomeWork
    wait a minute
end repeat
```

- Added ability to **wait until** *time* to allow a script to pause execution until a particular date/time arrives. If the time specified is already in the past, execution proceeds immediately.

```
wait until "5:00 PM"
wait until now + 1 second -- evaluated repeatedly; will wait forever
wait 1 second -- do this instead
```

- Added a **sorted** operator that provides all the functionality of the **sort** command. As an operator, it produces a sorted value without changing the source value. The sorted operator takes the form "**value sorted options**" where options include the same options available for the **sort** command, plus "**as chunkTypes**" to sort the value as something other than a list of items.

```
put (99,3,2,5,32,36,45737,26) sorted -- (2,3,5,26,32,36,99,45737)
put randomList sorted in ascending order into sortedList
put "Eggplant is great" sorted as words -- "Eggplant great is"
put "Eggplant is great" sorted as words by the length of each
-- "is great Eggplant"
```

- Added the **asTextLimit** global property to set a limit on the size of internally-generated text representations of values. If a text representation is requested that exceeds this limit an exception is now thrown. This protects the system from possible crashes caused by extremely large values. The default limit is ten million characters.
- Added a **globallyUniqueString()** function to return a unique string each time it is called. The value is unique across systems, application instances, and individual calls to the function.
- Added some additional pieces of information returned by the **SystemInfo()** function, and by the **OSInfo()** function on Linux and Windows systems.
- Added the **predefinedVariables** global property. This is a property list containing the definitions of all of the predefined variables. Any changes made to this property will affect any subsequent use of an unassigned variable whose name is one of its keys. By modifying this property, predefined variables can be created or removed, or their value can be changed.

```
put st -- "st"
set the predefinedVariables.st to "SenseTalk"
put st -- "SenseTalk"
```

- Added some additional predefined variables, in the Common Symbols, Numeric Symbols, and Miscellaneous Symbols categories. To see all of the predefined variables and their values, simply execute this command:

```
put the predefinedVariables
```

- Added **increasing** and **decreasing** as synonyms for **ascending** and **descending** for specifying the order of a sort operation.
-

Bug Fixes / Tweaks:

- Fixed a discrepancy between big and little endian mobile devices which resulted in a pixel shift issue.
- Fixed a problem setting the maximum window size to not account for the toolbar size.
- Added encryption support for RealVNC Mobile servers.
- The **imageFound()** function will now throw an exception if called with only one parameter that is a number. If the first parameter is a number it is treated as the time to search, so at least one image must also be supplied. To search for an image whose name is a number without also giving a time parameter, include the image's file type extension
- Fixed an occasional problem cutting text from a script.
- Fixed a problem where changes made to the Text properties were not made available to the AHDB windows. Now changes made in preferences will cause the AHDB context to reload the default values.
- Fixed a problem that could occasionally lead the **remoteClipboard()** function called with a time parameter to throw an exception incorrectly indicating that no new value was available.
- Fixed a crashing/memory problem when setting the **SearchRectangle** repeatedly in a script.
- Fixed a bug in the cursor location toolbar which would report an incorrect value when the screen was scrolled.
- Fixed a rare hang when running a script from the command line.
- Fixed an issue setting the fonts used for scripts and logs.
- Fixed a problem with the "New Script..." menu item being disabled when a window other than a script or suite window was selected.
- Improved the behavior when typing the name of an image to save that is the same as (or begins with) the name of an image folder or collection.
- Fixed a memory leak of a socket, which could lead to a crash when checking connection availability.
- Fixed a problem with display of values in the Duration column on the Licenses panel.
- Fixed a bug with the Remove button on the Licenses panel deleting the last license rather than the selected license.
- Improvements to the UI behavior of all search fields.
- Fixed a bug that would throw an exception if **Add Image** was used to add an image to a commented line in a script.
- Fixed time values specified without a particular date to be treated as a time on the current day (previously a bug was treating these as being on the first day of the current year).
- Fixed a bug with deleting the first item of a list while iterating over that list (could cause the second item of the list to be skipped).
- Fixed a bug (introduced in 10.22) with displaying lists or property lists containing trees (including results of "all nodes of" expressions). The trees would appear as an empty property list.
- Fixed some problems with the **standardFormat()** function, which in some cases (such as values containing ">" followed by a return) would produce an incorrect representation that could not be turned back into the original value using the value() function.
- Fixed a problem with checking whether a range **is within** another range.
- Fixed a bug that would display ranges within lists or property lists as lists, rather than as ranges.
- Fixed a bug that would throw an exception if a **next repeat** statement was executed more than 10 times within a try block inside a repeat loop, with the message "SenseTalk Runtime Exception - TRY statement exceeds maximum nesting limit per handler (10)".
- Fixed a bug (introduced in 10.22) with displaying lists or property lists containing references to objects.
- Fixed a script formatting bug (introduced in 10.2) that would cause problems when pasting any odd-length text string at a location exactly 99 characters before the end of a script that is at least 200 characters long.
- Fixed a bug with standardFormat() that would throw an exception in certain cases (such as when a property list contained a standard Cocoa object).
- Improved the error message when the call depth limit is exceeded to provide a helpful hint. This error was also **renamed from "SRUN_RecursionLimitExceeded" to "STMaxCallDepthExceeded"**.
- Fixed a number of script memory leaks, including some when accessing chunks of a value.
- Fixed a crash when accessing lines of a file in certain situations.

- Fixed a crash when setting a variable to one of its items that is an integer.
 - Fixed a bug that could lead to a crash if an exception occurred within a sort condition inside a try block.
 - Fixed a problem with formatting of lists and property lists that would fail to change the quote format back after rendering a list within a property list or vice versa.
 - Added additional details to warning when unable to locate suite for a messenger.
-

Release v10.23 (10-Dec-2010)

Highlights

- Bug fixes, stability and performance improvements.
-

User Interface

- Fixed a crashing problem when deleting or replacing an image and then accessing images through the Images Tab.
 - Fixed a problem when deleting logs from the Script Results pane.
 - Fixed a problem using the Insert.. pop-up button to insert a RightClick command.
 - Cleanup of About Panel GUI.
-

VNC

- Fixed a crashing problem when using the Zlib based encodings. ZlibHextile, ZlibRaw, and ZRLE should all be more stable.
 - Reduced Remote Screen "Shimmering" when in Full Size mode -- this shimmer still occurs in scaled mode.
-

Scripting

- Added a **sorted** operator for beta testing. This takes the form "*value sorted options*" where options include the same options available for the **sort** command, plus "**as chunkTypes**" if you want to sort the value as something other than a list of items. This feature is considered beta, as it needs more testing and may undergo some finishing tweaks before the next release.

```
put (99,3,2,5,32,36,45737,26) sorted -- (2,3,5,26,32,36,99,45737)
put randomList sorted in ascending order into sortedList
put "Eggplant is great" sorted as words -- "Eggplant great is"
```

- Added **millionth** as a valid ordinal number.
 - Added **millisec**, **millisecs**, **microsec**, and **microsecs** as abbreviations for millisecond, milliseconds, microsecond, and microseconds.
-

Bug Fixes / Tweaks

- Fixed a problem creating TextPlatforms or TextStyles that only differed by case.

- Cleanup of preference panes GUIs.
 - Fixed some issues with storing data into byte chunks that would previously convert values to and from text internally.
 - Fixed the standardFormat function to format lists with a space after the opening brace so that nested lists can be evaluated correctly by the value function without being interpreted as block quotes.
 - Fixed the "number of lines of" function and "each line of" expressions to allow a list for "delimited by" and use the lineDelimiter if delimiter is omitted or empty.
 - Fixed a problem with error reporting when an error occurred in evaluating the setup conditions of a repeat loop, that would report as being an error in the preceding line of the script.
 - Fixed a bug with accessing the properties of an object that was passed into a handler as a single-item list containing the object.
 - Fixed a bug with accessing the currentIndex property or iterating using nextValue or nextAssignedValue of a list when that list contains a single value which is an object.
 - Fixed a bug with assigning a list of chunks to a value that would treat them as references ("put items (3,4) of x into y").
 - Fixed a bug when performing vector arithmetic using a list containing references to other values, that would alter the referenced values.
 - Fixed a problem where Remote Windows sometimes tracked the mouse when not the active window.
 - Fixed a problem with GetOptions reporting about FullScreenFailsafe.
 - Fixed a bug that prevented iterating over the same range or list multiple times in a nested manner.
 - Fixed a bug when saving a script to a different Suite.
 - Fixed a bug where non-active Remote Windows would report that they were running the script.
 - Fixed a bug that would prevent similar objects (that were equal in value, but distinct objects) from both being included in the helpers or early helpers of another object.
 - Fixed a crash when the SUT screen resolution changed.
 - Fixed a problem where enabling/disabling some encodings wouldn't take effect.
 - Fixed a bug using the TypeText pull-down menu.
-

Release v10.22 (9-Sep-2010)

Highlights

- Bug fixes, stability and performance improvements.
-

User Interface

- Fixed a problem entering a VNC password on RedHat based distributions.
-

VNC

- Added control to optionally exclude VNC screen synchronization using the defaults write Eggplant RequireSync NO
-

Scripting

- Improved performance when displaying large lists or property lists or converting them to text.
- Restored the default behavior of **AsTextProperty** to **enabled**.

- Added optional "indent" property to the listFormat and the propertyListFormat. Setting this property to a suitable string such as " " will cause individual list items or property list entries to be displayed on separate lines with values indented by multiples of the indent value according to the nesting of the structure.

```
set list to (apple, banana, cherry)
put list -- (apple,banana,cherry)
set the listFormat's indent to " " -- indent by 2 spaces
put list --
(
  apple,
  banana,
  cherry
)
```

Bug Fixes / Tweaks

- Fixed a hanging problem when viewing **ScriptResults()**.
 - Fixed a bug that prevented iterating over the same range or list multiple times in a nested manner.
 - Fixed a bug that would prevent similar objects (that were equal in value, but distinct objects) from both being included in the helpers or early helpers of another object.
 - Executions of **runscript** will once again block and report output until the scripts have finished executing.
-

Release v10.21 (12-Aug-2010)

Highlights

- Improved VNC performance and compatibility including KVM enhancements.
 - Major stability improvements.
-

User Interface

- You can set a Default search type for new images in the Viewer Preferences.
 - Results for scripts called with RunWithNewResults can now be easily jumped to by using the Linker Arrow in the Result Viewer.
 - The Format menu has been removed, as creating scripts in RTF format is no longer supported. Similarly, the Default Script File Format has been removed from the Script preferences pane.
 - The Restore Defaults action on the Text preferences panel now displays a dialog warning that it will remove any custom text platform information.
 - Added the ability to pick a SUT from the Connection List on the Schedule Tab.
 - A new command on the Help menu provides a link to the new online tutorials.
-

VNC

- Fixed a crashing problem when using the Zlib based encodings. ZlibHextile, ZlibRaw, and ZRLE should all be more stable.
- Added support for maintaining a connection when changing SUT screen resolutions.

Scripting

- The current setting of **the ScriptLogging** property is no longer reset by the **RunWithNewResults** command. The ScriptLogging property is now shared in common between the master script and any scripts called by the RunWithNewResults command, so any change to the property in one script will carry over to the others.
- Improved performance of **scriptResults()** function.
- Enhanced performance of the **format()** function. This function also now supports additional format codes %a, %A, and %@ (%a and %A are not available on Windows).

Bug Fixes / Tweaks

- Fixed sporadic crashing problems associated with entering protected code.
- Fixed a crash when the SUT screen resolution changes.
- Fixed an occasional crash when being prompted for a VNC password.
- Fixed a problem requesting a trial license before any license has been entered.
- Fixed a problem accessing Suites from a scripts which had not previously been opened.
- Fixed a bug with a scripted TIG not working correctly when the TIG script is not located in the current suite.
- Fixed a problem undoing a script change.
- Fixed a problem where enabling/disabling some encodings wouldn't take effect.
- Fixed a bug testing the existence of an object property when the property name is in a variable.
- Fixed a problem with some error messages (such as when attempting to access a URL) not reporting NSInvalidArgumentException ***-[NSCFTType stValue]: unrecognized selector sent to instance.
- Improved some error messages.
- Fixed a bug which would cause a crash when performing various operations on a list or property list that contained a circular reference to itself.
- Fixed a bug with deleting an item from a range.
- Fixed a bug with copying list items between lists or to a different place within a list, that would incorrectly treat the copy like a reference to the original item.
- Fixed a bug with passing a list item as a parameter to another handler, that would in effect pass it by reference.
- Fixed a bug with sorting items in a list nested within another list.
- Fixed a bug that would throw an exception ("STInvalidOperation -currentIterationIndex called on non-object STValue") when trying to iterate "by reference" over the items of a list (using "repeat with each item of theList by reference") when the list was received as a parameter or is itself a reference.
- Fixed a bug with **the exception** not being set to empty on a single-line **try** statement that doesn't throw an exception.
- Fixed a problem introduced in the compiler in release 1.36 (Eggplant 10.03) that would treat the plain words 'date' and 'time' as function calls rather than variables.
- Fixed a problem with forcing dates to compare as text in some situations ("the time as text is someTime" would compare as times rather than text).
- Always use normal SenseTalk formatting when storing structured values into a text chunk or a file.
- Fixed a problem with storing into files in some situations. This now correctly stores the respective values into the two files (previously it did nothing):

```
put ("abc","def") into (file "/tmp/a", file "/tmp/b")
```

- Fixed a bug that would return empty when using a range to access items in a list, such as:

```
put items 5..10 of 20..1
```

Release v10.2 (10-May-2010)

Highlights

- New Text Recording feature for Remote Window allows recording most key commands.
- Dramatically improved script formatting and editing performance.
- Major stability and User Interface improvements.

User Interface

- The New Text Recording feature allows you to just start typing when in Capture Mode in the Remote Screen Window. A panel showing the TypeText command to be inserted will allow you to preview your command before sending it to the SUT. This feature replaces the insert TypeText action.
- Menu actions now use the traditional Linux Ctrl-Key instead of the Meta-Key.
- You can set a Default search type for new images in the Viewer Preferences.
- You can set a Default Image Format for recorded images. Issue this command in a terminal using a valid TYPE: ("png", "tif" or "tiff"):
`default write Eggplant ImageFormat TYPE`
- You can now double-click in lists to open Connections, edit Scripts and more.
- Major performance improvements in script editing.
- Bug Submission Panel now provides feedback on if the submission was accepted.
- Improved Control over helper suites, allows setting Home and Default Suite directory relative paths by using tilde(~) and dot(.) respectively. Select a helper and then double-click the path to edit.

VNC

- Dramatically reduced the number of screen updates Eggplant requests during typical exchange with the VNC server.
- Added the ability to forgo full-screen updates before image searches fail. These updates are difficult for slower VNC servers and certain KVM devices to handle. To disable full-screen updates issue this command in a terminal:
`default write Eggplant FullScreenFailsafe NO`
- Added a **mouseRepositionSpeed** to allow setting the reposition to happen slowly, rather than jumping. Generally this is not recommended as sliding the mouse can cause side effects on the screen, but for KVM and other devices it is sometimes necessary.

Scripting

- Quoted lowercase strings used in the context of the auto-released **ShiftKey** will now be interpreted as upper-case letters by all VNC servers. Example: `TypeText shiftKey, "n"` will print a capital "N" previously it was dependent on the target VNC Server.
- When running in FullScreen mode **Ask** or **Answer** panels will now be displayed on top of the remote screen.
- **Line** chunks are now defined as chunks of text separated by any of a list of delimiter strings (as specified by the `lineDelimiter` property), rather than always being separated by return characters.

- Added the **lineDelimiter** local property and the **defaultLineDelimiter** global property to permit changing the set of strings used as delimiters between "line" chunks. Either property can be set to a list of delimiter strings. The initial setting of the defaultLineDelimiter is simply the return constant (also known as linefeed or LF) for compatibility with earlier versions. Setting the defaultLineDelimiter to empty is a shortcut to set it to the list (CRLF, LF, CR) to match any of the common line endings. The lineDelimiter property is local to each handler execution context (or frame) and is set to the value of the defaultLineDelimiter at the start of each handler.

The order of the delimiters is significant, with preference being given to delimiters earlier in the list. So setting the lineDelimiter to (CRLF, LF, CR) will match CRLF (a carriage return character followed by a linefeed) as a single delimiter rather than as two separate line delimiters.

- Added the **characterFiller**, the **wordFiller**, and the **lineFiller** global properties to give control over how strings are extended when putting values into non-existent (out of range) character, word, or line chunks.

The characterFiller is initially set to "." (a period), so putting something into a character position beyond the current value in a container will fill in the intervening positions with dots (NOTE: this is a change from earlier behavior):

```
put "A" into notes
put "G" into character 7 of notes
put notes -- "A.....G"
```

The characterFiller isn't restricted to a single character. A longer string will be repeated as needed to fill in the space:

```
set the characterFiller to "Ha"
set laugh to empty
put "!" into character 7 of laugh
put laugh -- "HaHaHa!"
```

The wordFiller is initially set to "?" to represent any unspecified words (NOTE: this is a change from earlier behavior):

```
set phrase to "start here"
put "the end" into word 7 of phrase
put phrase -- "start here ? ? ? ? the end"
```

Set the wordFiller to empty for the same behavior as earlier releases (simply fill in with a single space), or to a different value:

```
set the wordFiller to "umm..."
put "Hello and" into greeting
set word 5 of greeting to "welcome!"
put greeting -- "Hello and umm... umm... welcome!"
```

The lineFiller is initially set to **return** for the same behavior as earlier versions, but may be set to a different line ending (such as CR or CRLF) or some other text if desired.

- Added **top()**, **bottom()**, **left()**, and **right()** functions to get the location of the indicated edge of a rectangle (a single value).
- Added **center()**, **topCenter()**, **bottomCenter()**, **leftCenter()** and **rightCenter()** functions to get the point at the center of a rectangle, or at the center of the top, bottom, left or right edge, respectively, of a rectangle.
- Added the ability to test whether any value is an iterator using the "is an iterator" operator (lists, ranges, and objects with an objectType of "iterator" are all iterators).


```
put (1,3,5,9) is an iterator -- true
```
- Fixed a problem with multiple each expressions within an expression that could cause a hang. It is now possible to use more complex expressions that combine each expressions, like these:


```
put each item of "A.."C" & each item of 1..4
  -- ((A1,A2,A3,A4),(B1,B2,B3,B4),(C1,C2,C3,C4))
put each item of 1..10 times each item of 1..10 into timesTable
```
- Parentheses can now be used around "for each" expressions and "where" clauses to enhance readability and limit the scope of the expression:


```
put each.first && each.last (for each item in employeeList)
put each line of sales (where item 3 of each > 5000)
```

- Added **is at least** and **is at most** as synonyms for the **greater than or equal to** and **less than or equal to** operators. Also added **is no less than** and **is no more than** variants for these operators:

```
set x to 5
put x is at least 5 -- true
put x is no more than 7 -- true
put x is at most 5 -- true
put x is no less than 5 -- true
```
 - Assigning values to chunks with out-of-range negative indexes now properly inserts chunks at the beginning of the container as necessary:

```
set list to (1,2,3)
put "X" into item -6 of list
put list -- (X,,1,2,3)

put "A" into goingDown
put "G" into character -7 of goingDown
put goingDown -- "G.....A"
```
 - Changed name of the initialHandler script in a folder object from "<initialHandler>" to "_initialHandler_" (for greater compatibility across different file systems).
-

Bug Fixes / Tweaks

- Fixed problems with script execution hanging when running Command Line (runscript).
 - Selecting text within a script now works when dragging beyond the window region.
 - You can now use the Open Script dialogue to open scripts inside any Suite.
 - You can now select any text in the Run Window (Script or Log).
 - Fixed a problem calling ConnectionInfo() on a server that didn't exist - this now returns an empty property list.
 - Fixed a problem using the Connect() commands which could result in an exception.
 - Fixed a problem where the Connect() command was case-sensitive on properties that were passed in.
-

Release v10.12 (09-November-2009)

Linux

Highlights

- New Text Recording feature for Remote Window allows recording most key commands.
 - Images are now stored in PNG format for compatibility to all versions of Mac OS X.
 - **Text Platforms are now stored in the file `~/Library/Eggplant/TextPlatforms.plist`. See below for more detail.**
-

User Interface

- The New Text Recording feature allows you to just start typing when in Capture Mode in the Remote Screen Window. A panel showing the TypeText command to be inserted will allow you to preview your command before sending it to the SUT. This feature replaces the insert TypeText action.
- Added the ability to enable/disable caching at the TextPlatform level.
- Added the ability to reset TextPlatforms to their default values.
- Clearer enabling/disabling of controls when doing Image Doctor searches.
- Improved Suite Info Panel

Scripting

- When running in FullScreen mode Ask or Answer panels will now be displayed on top of the remote screen.
- **Line** chunks are now defined as chunks of text separated by any of a list of delimiter strings (as specified by the `lineDelimiter` property), rather than always being separated by return characters.
- Added **the `lineDelimiter`** local property and **the `defaultLineDelimiter`** global property to permit changing the set of strings used as delimiters between "line" chunks. Either property can be set to a list of delimiter strings. The initial setting of the `defaultLineDelimiter` is simply the return constant (also known as linefeed or LF) for compatibility with earlier versions. Setting the `defaultLineDelimiter` to empty is a shortcut to set it to the list (CRLF, LF, CR) to match any of the common line endings. The `lineDelimiter` property is local to each handler execution context (or frame) and is set to the value of the `defaultLineDelimiter` at the start of each handler.

The order of the delimiters is significant, with preference being given to delimiters earlier in the list. So setting the `lineDelimiter` to (CRLF, LF, CR) will match CRLF (a carriage return character followed by a linefeed) as a single delimiter rather than as two separate line delimiters.

- Added **the `characterFiller`**, **the `wordFiller`**, and **the `lineFiller`** global properties to give control over how strings are extended when putting values into non-existent (out of range) character, word, or line chunks.

The `characterFiller` is initially set to "." (a period), so putting something into a character position beyond the current value in a container will fill in the intervening positions with dots (NOTE: this is a change from earlier behavior):

```
put "A" into notes
put "G" into character 7 of notes
put notes -- "A.....G"
```

The `characterFiller` isn't restricted to a single character. A longer string will be repeated as needed to fill in the space:

```
set the characterFiller to "Ha"
set laugh to empty
put "!" into character 7 of laugh
put laugh -- "HaHaHa!"
```

The `wordFiller` is initially set to "?" to represent any unspecified words (NOTE: this is a change from earlier behavior):

```
set phrase to "start here"
put "the end" into word 7 of phrase
put phrase -- "start here ? ? ? ? the end"
```

Set the `wordFiller` to empty for the same behavior as earlier releases (simply fill in with a single space), or to a different value:

```
set the wordFiller to "umm..."
put "Hello and" into greeting
set word 5 of greeting to "welcome!"
put greeting -- "Hello and umm... umm... welcome!"
```

The `lineFiller` is initially set to **return** for the same behavior as earlier versions, but may be set to a different line ending (such as CR or CRLF) or some other text if desired.

- Added **`top()`**, **`bottom()`**, **`left()`**, and **`right()`** functions to get the location of the indicated edge of a rectangle (a single value).
- Added **`center()`**, **`topCenter()`**, **`bottomCenter()`**, **`leftCenter()`** and **`rightCenter()`** functions to get the point at the center of a rectangle, or at the center of the top, bottom, left or right edge, respectively, of a rectangle.
- Added the ability to test whether any value is an iterator using the "is an iterator" operator (lists, ranges, and objects with an `objectType` of "iterator" are all iterators).

- ```

 put (1,3,5,9) is an iterator -- true

```
- Fixed a problem with multiple each expressions within an expression that could cause a hang. It is now possible to use more complex expressions that combine each expressions, like these:
 

```

 put each item of "A".. "C" & each item of 1..4
 -- ((A1,A2,A3,A4),(B1,B2,B3,B4),(C1,C2,C3,C4))
 put each item of 1..10 times each item of 1..10 into timesTable

```
  - Parentheses can now be used around "for each" expressions and "where" clauses to enhance readability and limit the scope of the expression:
 

```

 put each.first && each.last (for each item in employeeList)
 put each line of sales (where item 3 of each > 5000)

```
  - Added **is at least** and **is at most** as synonyms for the **greater than or equal to** and **less than or equal to** operators. Also added **is no less than** and **is no more than** variants for these operators:
 

```

 set x to 5
 put x is at least 5 -- true
 put x is no more than 7 -- true
 put x is at most 5 -- true
 put x is no less than 5 -- true

```
  - Assigning values to chunks with out-of-range negative indexes now properly inserts chunks at the beginning of the container as necessary:
 

```

 set list to (1,2,3)
 put "X" into item -6 of list
 put list -- (X,,1,2,3)

 put "A" into goingDown
 put "G" into character -7 of goingDown
 put goingDown -- "G.....A"

```
  - Changed name of the initialHandler script in a folder object from "<initialHandler>" to "\_initialHandler\_" (for greater compatibility across different file systems).
- 

## Bug Fixes / Tweaks

- Text Platforms are no longer part of the preferences (aka defaults) system and are now stored in the file `~/Library/Eggplant/TextPlatforms.plist`. When 10.1 first launches it will relocate your Text platform preferences to there.
  - Fixed a bug where the first call of an image in a scripted TIG as a Text Image Generator was ignored.
  - Fixed a bug where image search type was reset when doing Image Doctor: Replace Image.
  - Fixed a bug where Eggplant could hang from Command Line when Auto Doctor was enabled.
  - Fixed a bug when making a collection of an image with a DOT in it's name.
  - Fixed a problem passing file names into the PLAY command.
  - Fixed a host of problems entering and storing information in the Text Preferences.
- 

## Release v10.05 (04-September-2009)

### LINUX

#### Highlights

- Easier startup commands on LINUX
- LINUX now ships with the Eggplant Floating License Server.
- Fixed many a number of bugs and other stability issues.

---

## User Interface

- The Eggplant Floating Licenser application is now included for users with floating licenses.
  - The Eggplant Package Now Installs the following Shell Commands:
    - eggplant** - to launch eggplant and log the output.
    - efls** - to invoke the eggplant floating license server.
    - defaults** - to alter user preferences when required.
    - eggplant-debug** - to launch eggplant using GDB when requested by TestPlant support.
- 

## Bug Fixes / Tweaks

- Setup Eggplant installer to run on SELinux enabled systems.
  - Fixed warnings raised by the installer about setting a partial context.
  - Fixed a bug setting breakpoints and typing text on the first line of a script.
  - Eggplant output on LINUX is now automatically stored in ~/Library/Logs.
  - Fixed a bug changing the default suite directory on LINUX.
  - Fixed a bug on LINUX setting text generators in the Text Prefs pane.
  - Fixed a hanging bug on LINUX when processing UserDefaults -- commonly hit when reconnecting SUTS on launch.
  - Fixed a hanging bug on LINUX as a result of xwindows exceptions.
  - Fixed a crashing bug on LINUX when setting remote window title -- usually when starting a script.
  - Fixed crashing bugs on LINUX when hitting xcb\_lock exceptions.
- 

## Release v10.04 (20-August-2009)

### LINUX

### Highlights

- Eggplant preferences (aka defaults) are now stored under the Name **Eggplant** for both Linux and Mac platforms. Eggplant will continue to honor the old setting at **com.redstonesoftware.Eggplant** but the new **Eggplant** domain will take precedence.
  - Fixed a number of bugs and other minor issues.
  - Includes Vine Server 3.11 which fixes problems with ZLibHexfile encoding.
- 

## User Interface

- Added the ability to reset script colorization to the default values.
  - Improved the Licensing Panel to allow specifying the Host Address for a Floating License Server.
- 

## Command Line Interface

- Added a **GlobalResultsFolder** *folderName* flag to request that all script results be written to a particular folder (or a particular folder relative to the suite).

---

## Bug Fixes / Tweaks

- Fixed a hanging bug on LINUX when processing Rich Cursor updates from some VNC servers.
- Fixed a hanging bug on LINUX when handling an abnormal termination from some VNC servers.
- Fixed a crashing bug on LINUX when receiving the Alert Bell from VNC servers.
- Bulletproofing on LINUX to prevent exceptions in sub-threads from shutting down the application.
- Bulletproofing on LINUX to prevent a deadlock when accessing preferences from sub-threads.
- Fixed a bug on LINUX where the capture window could get progressively darker.
- Fixed a bug on LINUX where the script log window would not stay scrolled to the bottom when new log information was displayed.
- Fixed a bug on LINUX where scripts called via the command line interface using a relative script filename did not reference results and other scripts properly.
- Fixed a bug on LINUX with failures to lock the X-Windows graphic context (xcb-lock errors).
- Eggplant for LINUX now uses the modern FFI library (instead of FFCall library) for communication.
- Fixed a bug where the name reported by the ConnectionInfo function would not change to reflect the display name specified in the Connection List.
- Removed log warnings for missing fonts.
- Added backtrace report output when Eggplant crashes.

---

## Release v10.01 (27-April-2009)

### Highlights

- Installer will now setup Eggplant to be launched from the Gnome->Applications menu system.
- Fixed a number of bugs and other minor issues.

---

## Bug Fixes / Tweaks

- Fixed a common problem where Eggplant wouldn't recognize changes in the "active script" resulting in an inability to record images in the current script.
- Fixed a crash that could occur on making a connection (either from a script or the GUI panel)
- Fixed a hang that could occur when clicking on the color well.
- Fixed a number of problems with menu's not enabling/disabling or updating titles.
- Fixed a problem reopening scripts/suites.
- Fixed a problem with Eggplant not opening certain documents (Getting Started/Release Notes)
- Fixed a problem customizing the Viewer Window Toolbar.

---

## Release v10.00 (20-April-2009)

This initial release of Eggplant for Linux contains almost all of the functionality available in Eggplant for Mac. Below are the features that are excluded at this time. Please report any problems that you encounter to [support@testplant.com](mailto:support@testplant.com).

---

## Known Exclusions from the Mac version of Eggplant:

- **Movie Recording** both from the GUI and the **startMovie** command is not supported on Linux.
- **XML** syntax with **tree** and **node** within SenseTalk is not support on Linux.
- **Text Image Generation** is only supported through **external** and **scripted** generators.
- **Bonjour/zeroconfig** VNC server detection is not present.
- **sendmail** command does not support the **Cram-MD5** authentication style.
- **Color Panel** Does not currently support the **Color Picker (Magnifying Glass)** or **Named Colors**.
- **ssh** connections ARE supported, but many LINUX versions restrict the ability to pass in the SSH password. We encourage you to set Public-Key Encryption to ensure that your SSH connections work.
- **Redo** is not implemented in the Script editor.

*Copyright (C) 2002-2011 TestPlant, Inc – All Rights Reserved  
US Patent #7,870,504. Other patents pending.  
Eggplant is a trademark of TestPlant, Inc.*