



iViewer for iOS Release Notes

For release notes specific to the Android version of iViewer, please see [iViewer for Android](#).

Version 5.0.10 (build 414)

Date Released: TBA

- Yet another fix for WiFi SSID access in iOS 12/13

Version 5.0.8 (build 410)

Date Released: October 5th, 2019

- Fix for WiFi SSID access in iOS 13

Version 5.0.7 (build 409)

Date Released: September 20th, 2019

- NEW icon to comply with app store guidelines
- Fix potential crasher when closing remote debugger dialog

Version 5.0.6 (build 407)

Date Released: September 16th, 2019

- Fixes for single page licensing when resuming app from background

Version 5.0.5 (build 406)

Date Released: September 13th, 2019

- Fixes for app links to allow loading new [GUI](#) when app was not already open in background
- Progress bar for [GUI](#) loading and unzipping status
- Allow [app links](#) to enable debugging using `remoteDebugging=1` and `remoteDebuggingPort=XXX` in the URLs.

Version 5.0.1 (build 400)

Date Released: September 6th, 2019

- Major version bumped and released as new app
- Licensing changes required a new app to avoid current iViewer 4 installs experiencing licensing problems in future releases.

- Backwards compatible with iViewer 4 projects
 - The Device ID generated will be different than iViewer 4. For this reason, existing licenses will not work in iViewer 5. You can transfer your license to the new ID that iViewer 5 generates.
-

Version 4.1.393

Date Released: July 31st, 2019

- Support for latest iOS device resolutions (Except iPad Pro, fix coming soon!)
- Support for multiple GUI files in a single zip archive.
- Added support for iBeacon detection via JS API - see Sensors JS API [<http://www.commandfusion.com/docs/scripting/sensors.html>]
- Support for overriding theme properties (text size, color and alignment) on buttons and text objects, without having to create new themes. Install guiDesigner v2.6.7.0724 [<http://commandfusion.com/software/guidesigner/history>] to test it out!
- Adds CF.playSound and CF.stopSound JS APIs - CF.playSound("sound name");
- Fix to allow HTTP Client systems to use SSL (<https://>)
- Fix setting/getting input field values via tags
- Fix for using tags via CF.listAdd calls
- Fix for tags starting with 'd' and 'a' on text objects
- Allow zips to be loaded without requiring a .zip extension in the GUI File URL

Version 4.0.353

Date Released: May 29th, 2017

- Text wrapping issues resolved (introduced in build 351)

Version 4.0.351

Date Released: April 21st, 2017

- 64 bit support (remove pesky Apple popup)
- Updated math expression library
- Autoscaling for smaller GUI projects on larger devices (iPhone 5 projects will scale to fit iPhone 6/7)

Version 4.0.319

Date Submitted: TBA

- Improved iOS 8 compatibility
- Improved Javascript support
- Added new Javascript CF.ApplicationCallbackEvent to support passing data to JS from external applications
- Added new Javascript CF.launch() API
- Added new callback option for CF.openURL API (to report whether opening an URL succeeded)
- Support fallback to previous GUI in case a GUI update fails
- Prevent device from sleeping while playing any media (local video, local audio or remote AirPlay)
- Fixed issue where trying to open an invalid URL (i.e. using CF.openURL) could crash iViewer
- Fixed issue with CF.loadGUI not loading the default GUI when passed an empty string
- Fixed issue with JS not receiving proper network information after a GUI switch
- Fixed status bar appearance issue when set to CF.STATUS_BAR_BLACK_TEXT

Version 4.0.303

Date Released (iViewer 4): April 14th, 2014

- Improved application memory handling so as to reduce the chances for the OS (in particular iOS 7) jettisoning the app when it needs to claim memory
- Added a `tags` local variable to the JS execution context for buttons and gestures, giving the code access to the calling object's tags
- Do not let the device go to sleep while playing video through AirPlay
- Fixed issue with lists where footer subpage could be the wrong size
- Fixed issue with the Stop Macro / Stop All Macros feature
- Fixed issue with high density images (@2x) images that could lead to a slider thumb doubling in size
- Fixed an issue with `CF.networkStatusChangeEvent` not always being fired as it should
- Fixed issue with input fields background color on iOS 7

Version 4.0.296

Date Released (iViewer 4): March 4th, 2014

- Added new `CF.networkType` for AdHoc connections reported to JS (“AdHoc”).
- Fixed an issue where iViewer would not detect that the WiFi network is active when the device is connected in AdHoc mode.
- Fixed `CF.loadAsset()` constants (`CF.CACHE` and `CF.NO_CACHE` values were swapped, resulting in `CF.CACHE` not working as expected)
- Fixed `CF.unwatch()` calls when un-watching a watch that had specific parameters

Version 4.0.292

Date Submitted (iViewer Next): January 7th, 2014

- Improved `CF.listContents` to now return all values and items, including those not backed by actual GUI elements in the list
- Implemented new `CF.StatusBarAppearanceProperty` for `CF.setDeviceProperty`, with accepted values `CF.STATUS_BAR_HIDDEN`, `CF.STATUS_BAR_WHITE_TEXT` and `CF.STATUS_BAR_BLACK_TEXT` (last two values produce the same result, showing the status bar, on OS versions prior to iOS 7)
- Fixed networking issues with Bonjour browsing, iViewer Protocol, and TCP SSL connections
- Fixed iOS 7 issues with text field placeholders not always displayed and URLs with certain non-ascii characters not being URL-encoded the same way they were in previous releases.

Version 4.0.288

Date Release (iViewer): October 3rd, 2013

- Improved iOS 7 compatibility
- Fixed several crashes in various areas of the product
- Fixed JS issues with `CF.listContents()`, `CF.addTag()`, `CF.removeTag()`

Version 4.0.282

Date Release (iViewer): September 24th, 2013

- iOS 7 compatibility
- New **tags** support: every object in the GUI that carries a join can be augmented with arbitrary tags attached to it. Any time a join is referenced, tags can be substituted. Uses include grouping radio buttons, grouping objects for easily changing their properties, accessing objects in a generic fashion from modules that don't need to be tied to specific join numbers, etc.
- New multiple control systems: iViewer now supports multiple “control” systems (systems that talk the iViewer protocol and receive join updates). As a side effect, the single “control system” is now going away from guiDesigner and automatically migrated to being placed in the overall list of systems. This also provides better consistency for newcomers.
- New options for TCP client systems: 'Dialog' mode (send data and expect reply before disconnecting) with dialog timeout, and idle timeout for non-always-on systems.
- Improved networking reliability
- TCP server systems now execute their startup command/macro on each incoming connection, and now properly honor the *Queue messages when offline* option
- Several fixes to JS API

Version 4.0.263

Date Release (iViewer): June 19th, 2013

- Fixed issue when status bar was visible when switching between GUIs

Version 4.0.262

Date Release (iViewer Lite): May 27th, 2013

- Networking improvements with faster, more reliable restart after sleep
- Added support for https connections to sites with self-signed certificates in web view
- Fixed issues with sliders, networking, Bonjour, status bar, subpage hide/show animations, video stop trigger

Version 4.0.254

Date Release (iViewer Next): April 4th, 2013

- Added new `CF.networkType` and `CF.networkSSID` JavaScript variables to help determining which network the device is on. Global tokens `[networkType]` and `[networkSSID]` were also added.
- Improved overall network performance: faster and more reliable reconnections, more accurate network status detection, better handling of multiple simultaneous `CF.request()` [<http://commandfusion.com/docs/scripting/net.html#cf.request>] calls
- Improved caching support with cache and pre-caching control via `CF.loadAsset()` [<http://commandfusion.com/docs/scripting/util.html#cf.loadAsset>]. New options have been added to precisely control the cache.
- Fixed issues with MJPEG streams display, sliders, screen rotation when status bar is displayed, and more.

Version 4.0.242

Date Released: February 20th, 2013

JavaScript

- New `CF.loadGUI(url, options)` API
- New `CF.device.uniqueIdentifier` and `CF.device.legacyUniqueIdentifier` properties
- `CF.request()` can now take no callback function (or a null callback) if you're not interested in the reply
- `CF.request()` now accepts any method for an HTTP request, not just the base ones defined in the HTTP RFC. Allows

easier integration with various protocols running on top of HTTP

- CF.getGuiDescription() now returns header, footer, title and contents subpage names as part of the description of a list object
- New CF.guiURL variable containing the URL of the current GUI
- CF.setProperties() can now update properties of joins in list items, using the standard list notation (listjoin:index:join)
- CF.watch() now outputs meaningful information to debugger log about invalid parameters
- New Sensors API to access and read some device sensors. You can receive updates for the device's accelerometer, gyroscope, magnetometer and GPS so as to obtain acceleration, rotation, heading and location data
- Gestures now provide the same context as buttons when executing associated JavaScript (join, list, listIndex and tokens variables in addition to the gesture variable)
- Fixed issue with properties rotation around Y axis

Networking

- New options for UDP systems to accept packets coming from any host and/or any port
- Fixed networking connection and reconnection on wakeup issues
- Fixed UDP issues with broadcast and multicast sockets
- Fixed issue with receiving packets over UDP broadcast systems
- Ensure loopback systems can run even when WiFi / 3G is turned off or disconnected.
- Commands that do the optional value assignment were not unescaping their data when assigning to a list (assigning to a list is using the list protocol and requires some binary separators in the data)

General

- Can now specify fonts to use with a platform (iOS, Android) font selector. Requires guiDesigner 2.5.0.0 or later.
- Added "[uuid]" global token with device UUID
- Better management on alerts, in particular connection loss / asset load warnings
- Fixed critical issue with buttons in lists
- Fixed text wrapping in labels with carriage return characters
- Fixed issue with displaying subpages in the wrong order in some cases
- Support for iOS 6 and full iPhone 5 resolution
- Fixed issue where lists in subpage could improperly animate their content on first appearance
- Fixed issue with new installs where the multitasking flag looks turned ON but really is not until the settings are changed once.
- Fixed several minor issues
- Fixed issue with slider thumbs not being set their slider's opacity when using CF.setProperties()
- Lists-in-lists (using headers and footers) are working again, even though not officially supported
- Ensure join s0 is never set as the result of triggering an associated digital join (i.e. input field on join 0 with an associated digital join, triggering the digital join would set the contents of the input field to all objects on s0 in the GUI)

Version 4.0.215

Date Released: June 15th, 2012

Changes

- Now issue a sensible log message through CF.log() if your JS tries to CF.send() before the end of the CF.userMain startup sequence
- Improved the connection speed to TCP systems in some cases
- Added new CF.ListWillStartScrollingEvent, CF.ListDidScrollEvent, CF.ListDidEndScrollingEvent for JS code to precisely know when a list scrolls
- Tweaked JS code execution engine so that some JS code can be executed while a list is scrolling

Fixes

- Fixed issue loading directory structures in zip files
- Fixed possible issue with lists if using incorrectly defined subpage names
- Fix connection authorisation with blank passwords
- Fixed issue with authenticated MJPEG feeds: wasn't authenticating if password was empty
- Reverted an earlier change that is messing up with URL strings provided by CF.request()
- Allow CF.LastItem to be used in CF.listScroll() for absolute positioning
- Fixed issue where slider's digital “pressed” join was being sent directly to main control system. Wasn't going through the internal join routing mechanism - no event fired to JS (confused users of CrestronMobile)
- Fixed a crash case where an invalid group number if used in a feedback item group reference
- Fixed an issue where in some cases, iViewer would fail reconnecting a TCP system
- Fixed grammar in log messages issued by the remote debugger
- Fixed issue where changing the theme of a gauge or slider using CF.setProperties() wouldn't work
- Disabling a system (using CF.setSystemProperties()) wasn't being honored if the system never connected and was still trying to connect
- Fixed an issue with images on the same join not always staying in sync if the serial join value (image URL) changed, accross page flips or orientation changes
- Fixed an issue where subpages disappearing to left or top would appear to disappear faster than their programmed transition
- Fixed issue where CF.openURL() would only work once
- Fixed issue where main control system would be created as a loopback system if its IP address was initially set to 0.0.0.0 in the GUI XML
- Fixed an issue in recent device identity changes. Device identity should now be stable
- Fixed a couple hang and crash cases
- Fixed an issue where scroll commands and macros would not fire when performing programmatic scrolling (i.e. CF.listScroll())

Version 4.0.207

Date Released: April 3, 2012

New Features

- Added new CF.BatteryLevelProperty and CF.BatteryChargeStatusProperty properties for CF.DevicePropertyChangeEvent, allowing full monitoring of plugged / unplugged state and battery charge level in JavaScript. See <http://www.commandfusion.com/docs/scripting-beta/changes.html> [http://www.commandfusion.com/docs/scripting-beta/changes.html] for a list of changes. Also added new CF.device.batteryLevel and CF.device.batteryChargeStatus values in the CF.device object to reflect current values.

Version 4.0.206

Date Released: April 2, 2012

New Features

- Added new CF.SoundOutputVolumeProperty for CF.DevicePropertyChangeEvent, and new CF.device.soundOutputVolume variable (read-only) that contains the current hardware output volume, a float number between 0.0 and 1.0.

Version 4.0.205

Date Released: April 2, 2012

Changes

- Switching away from Apple's unique device identifier (UDID) which is deprecated in iOS 5 and is starting to cause app rejections. We now use a new, CommandFusion-generated identifier for each device. Compatibility with existing jobs is being preserved: if at first launch the app sees it's not using the default GUI, it sets itself up to use Apple's device identifier. We will be documenting the migration process in a separate document.
- Fixed an issue where CF.setJoin() could not trigger “system features” like Show / Hide status bar (joins d17931 / d17932).

Version 4.0.204

Date Released: March 27th, 2012

New Features

- Added a preference to enable all systems disconnection on app exit / device sleep. By default, this pref is turned off
- Added new CF.device.screenBrightness property readable from JavaScript
- Added new CF.setDeviceProperty() call, allowing to change the device brightness
- Added new CF.DevicePropertyChangeEvent, allowing to monitor device property changes (currently, brightness only)

Fixes

- Fixed a lockup on app suspend (that would materialize as a restart of the app next time you used it)

Version 4.0.203

Date Released: March 25th, 2012

Submitted to App Store: NA

Fixes

- Tuned networking code for connectivity issues
- Make sure hostname-based systems' IP address get refreshed when needed
- Fixes to connectivity issues after inactivity periods
- Fixed alert that appears if the main control system's password is invalid (was appearing repeatedly)
- iViewer now shuts down all connections when it is suspended, and reactivates them when coming back. Since iOS 5 now abruptly disconnects connections on power off, this safety measure ensures that connections are not left in a half-closed state from the remote's perspective.

Version 4.0.201

Date Released: March 20th, 2012

Submitted to App Store: NA

Fixes

General

- Fixed issues reading GUI files that contain a BOM (byte order mark, added by some text editors)
- Fixed several instances of improper URL encoding / decoding (i.e. in button URLs, etc)

- GUI loading: HTTP redirects now carry our custom headers in redirected requests
- GUI loading: fixed issue with HTTP redirects to certain URLs that ended up with iViewer trying to load assets from the wrong URL
- Sounds: fixed an issue where cached sound files were not always made accessible after a restart

User interface

- Buttons now properly revert to their inactive state (when simulated) on page flip
- Update button text and state when associated joins update
- Fixed text wrapping (wasn't being honored in several cases)
- Fixed a crash in MJPEG processing
- Fixed an issue where list scroll commands and CFlistScroll() issues to a list on a hidden subpage would not be taken into account

Network

- Fixed issue where image loading, CF.request() and MJPEG video streams were altogether limited to 4 simultaneous requests. iViewer now allows 16 simultaneous CF.request() calls, and 16 simultaneous video streams.
- Fixed issue where UDP system didn't properly reconnect after a connection loss on forced sleep
- Startup command wasn't performing global token replacement, and can be sent to systems other than the one that just connected
- Fixed a crash in HTTP request processing
- When iViewer comes back from suspended / background state (after spending more than 10s there), requery the main control system for all its join states

Version 4.0.197

Date Released: February 7th, 2012

Submitted to App Store: NA

Fixes

- Fixed loading of certain MJPEG stream formats (Mobotix)
- Fixed CF.currentOrientation updates in JS
- Fixed OrientationChangeEvent callbacks in JS
- Fixed bug where CF.systems information would not be updated after using CF.setSystemProperties in some cases
- Fixed issue where CF.listAdd and CF.listUpdate would not affect all lists on the same join
- Added 'subpage' property to SubpageRef objects in CF.getGuiDescription callbacks to include the subpage name

Version 4.0.196

Date Released: Wednesday, 14th December, 2011

Submitted to App Store: Tuesday, 13th December, 2011

Fixes

- Fixed crash when loading .zip files containing JS
- Added migration code to allow updating from 4.0.5 without requiring a full GUI reload for cached GUIs
- Bumped GUI file load timeout from 10 seconds to 30 seconds
- Fixed broken CF.runMacro() JavaScript function

Version 4.0.194

Date Released: Tuesday, 6th December, 2011

Submitted to App Store: Tuesday, 29th November, 2011

iViewer Next Date Released: Thursday, 17th November, 2011 (build #190)

iViewer Next Submitted to App Store: Thursday, 10th November 2011 (build #190)

New Features

JavaScript

- New `CF.setSystemProperties()` [API](#) to change a remote system properties at runtime
- New scroll type '`CF.VisiblePosition`' for `CF.listScroll()` to allow scrolling a specific list item into view
- Added ability to set properties of objects within new list data when using `CF.listAdd`, same options as `CF.setProperties`
- Added the ability to use any subpage as a list item at runtime, instead of limiting to just title and content subpage types
- Added new '`xrotation`', '`yrotation`' and '`zrotation`' properties to an object's properties to allow getting and setting any object's rotation on the 3 axis (see `CF.getProperties()` and `CF.setProperties()` documentation)
- Added `CF.loadAsset()` to allow loading data from cache or new data files and receiving their byte contents in the callback.
- Added support for loading new GUIs via `CF.openURL()` calls using `cf://` scheme.
- New `CF.KeyboardUpEvent` and `CF.KeyboardDownEvent` to let JS code be notified whenever the soft keyboard appears or disappears
- New `CF.PreloadingCompleteEvent` for JS to be notified when the preloading / caching phase at [GUI](#) startup has completed

Networking

- Added support for customising HTTP/HTTPS requests sent out for images (img, buttons, sliders, etc) via tokens. Before loading an image that has an "`http://`" or "`https://`" [URL](#), iViewer will look at tokens whose name start with "`HTTP:`" and add them to the request headers (excluding the `HTTP:` prefix).
- Added proper text encoding support for systems via a new `textEncoding="..."` attribute in the XML, taking IANA encoding names (default is ISO-8859-1)
- Added ability to make any UDP system listen to incoming broadcast packets, regardless of its IP address (previously only broadcast systems would listen to incoming broadcast data).
- Added support for receiving base64-encoded HTTP replies.

GUI

- Clip contents of subpages to list bounds
- Added `scrollOffset` token to lists, giving the exact scroll offset of the list in pixels.
- Added new "`Middle`" option for list transitions (not available in `guiDesigner` yet).
- Added support for [URL](#) redirections for [GUI](#) loading, including zip files.
- Added '`clip`' attribute in XML for subpages, determines whether a subpage clips (`clip="1"`) or doesn't clip (`clip="0"`) its contents to its bounds. Note that regardless of this attributes, taps in subpage objects can be accounted for only within the subpage bounds.
- Added support for web views to run [HTML](#) embedded in the [GUI](#) assets (via zip file)
- Added support for cross-references of feedback capture group names in onmatch targets.
- Added pixel scroll position to `CF.listInfo` JavaScript callback.

Remote Debugger

- Support remote debugging of unencrypted zip projects.

Issues Fixed

JavaScript

- CF.GUISuspendedEvent will now be received by your JS code by the time the GUI is suspended. Some code can be executed at that time, but not too much (the application is going to background)
- Join changes via CF.setJoin() and CF.setJoins() are now propagated through the Control System connection if one is defined.
- Ensure JavaScript support is enabled even when no scripts are loaded, so that code attached to buttons/gestures/etc will still work.
- Performance improvements
- Changing an object's theme using CF.setProperties() wasn't working properly.
- CF.logObject() was not working properly in all cases
- CF.getGuiDescription() now correctly lists subpages and their contents
- Calling CF.request() with 3 parameters wasn't sending the request
- Fix bug where serial joins on buttons would not update correctly via JS
- In some cases, sending binary data with CF.send() wasn't working
- Remote monitor wouldn't reflect a systems address change through CF.setSystemProperties.
- Added 'tokens' variable to the JavaScript context when executing the code attached to a button (was documented but not working).
- Fixed issue with out-of-order callbacks (mostly at startup) for CF.ConnectionStatusChangeEvent
- CF.setProperties() now correctly updates button themes
- Fixed JS issue where CF.ConnectionStatusChangeEvent couldn't be watched for the main control system
- Fixed some JS issues with CF.watch() initial event (with last 'true' parameter) not always being fired correctly

Networking

- UDP port sending issues fixed.
- Fix for TCP Server mode to not close the server when last client disconnects.
- Fix crash when more than the allowed clients connect to an iViewer TCP Server
- Fixes for SSL HTTP Requests.
- Fixed broken HTTP redirection support for GUI download (was broken in 184 and up)

GUI

- Fix for subpage z-order issues
- Fix issue where slider indicators would render on top of all other content.
- Fix crash where slider indicator images could attempt to be rendered before being fully loaded.
- Fix rendering of slider active state when slider size is scaled up or down from the original image size.
- Fix crash when flipping pages whilst list is being filled.
- Fix bug relating to list item indexes becoming out of sync.
- Fixed bugs relating to rendering images with 0 height or width.
- Fixed “stretchy” buttons for Retina displays (@2x images)
- Fix for auto-refresh images in list header/footer subpages.
- Fix a bug relating to font bold and italics sometimes not being applied correctly.
- Various fixes for GUI File switching on button presses.
- Fix crash in MJPEG video object.
- Fix crash when going into background whilst changing GUI.
- Ignore page flips and subpage show/hide logic when digital join 0 receives updates.
- Fixes to video playback of cached video files.
- Fix issue where input fields would not send their commands when their digital join was raised.
- Fixed issues relating to disconnection join state.

- Fixed issues with keyboard layout selection (i.e. numerical) being lost while editing if device shakes
- Fixed issue with slider thumb sometimes showing behind slider itself
- Fixed issue with button text not updating when changed in list headers / footers (issue introduced in #188)
- Fixed issue introduced in previous builds where digital buttons in Header and Footer list wouldn't properly reflect their state

Remote Debugger

- Fix bug where images set to an empty path would still attempt to be loaded and throw an error if debug mode was enabled.
- Stop web view content loading until debug mode is ready (or cancelled).
- Fixed debug monitor connections for latest Chrome browser updates.
- Fixed spurious error reporting for cache loads with built-in GUI
- Fixed incorrect reporting in Remote Debugger where loading the built-in GUI would report load errors for images, even though they were here

General

- Fix crash when closing the app if using an asset folder.
- Fix token persistence, clearing persisted values when reloading a GUI.
- Fixes for macros including enforcing macro stopping and prevent loops when using macros within macros.
- Fix issues when reloading scripts when “Reload GUI Assets” or “Reload GUI Layout” is on.
- Fix issue where commands wouldn't be triggered if they contained only a space (and the JS calls were being ignored as a result).
- Fixed issue where bytes were generated in reverse order for math expressions.
- Fix when calculating list index via feedback processing.
- Fix for allowing app updates without reloading cache.
- Fix issue when stopping macros by macro name.
- Fixed issues with MJPEG video feeds not always auto-reconnecting after a disconnection or network loss
- General memory improvements
- added deltax and deltax tokens to Pan gesture, as well as .deltax and .deltay properties for JS info

Fixed for build 192 (not yet in app store)

- Fixed issue with CFloadAsset not finding cached assets
- Fixed issue relating to animating objects within lists
- Added log for remote address being connected to in debugger
- Fixed bug where CF.setSystemProperties would not always use the new port number
- Fixed bug where password and initialise messages were not being sent to control systems since build 185
- Fixed bug where the list index reported by items in a list could be incorrectly returned
- Added 'do not backup' flag for iOS5 cache to prevent the cache from being deleted when using iCloud

Version 4.0.5

Date Released: Friday, 29th July 2011

Submitted to App Store: Tuesday, 19th July 2011

Known Issues

- Gestures attached to page orientations function on either orientation rather than only the one defined

Changes

- Improvements to JavaScript bridge to increase performance
- JavaScript calls can now be made whilst a list is scrolling
- Changes to subpage handling on page flip - subpages no longer transition each time you flip (instead GUI is in same state on return flip)
- Added a range of custom headers to the GUI File URL request, including device ID, OS version, screen resolution, etc.
- Image objects and button overlay images will now render at their native image size if you specify zero for height or width.
- New CF.openURL method added to the JavaScript API
- Added button to continue loading GUI without debugging, even when debugging is enabled in settings
- Enforce scale limits to CF.setProperties (cannot go to 0 scale)
- Changes to the way objects are moved via CF.setProperties when their scale has been changed (use original scale for coords)
- CF.getProperties gives priority to onscreen items in cases where multiple objects exist on the same join
- Added CF.MACaddress and MACaddress token for accessing the device's MAC address
- Change to the way feedback on and off values for digital data parsing are implemented to allow exact onvalue matching without requiring offvalue

Fixes

- Fix crash when switching GUI files
- Fix issue where non-control system data was being interpreted as control system join data (issue with equals sign in data)
- Fix for remember last GUI file
- Fix for keyboard rotation issues
- Fix for sliders showing incorrect values in their indicator (off by 1)
- Fix crash when retrieving multiple properties at once via CF.getProperties
- Fixed saving and reloading of persistent global tokens
- Fix for video tool to stop loading data when hidden (if defined) and disable auto playing when not visible
- Fixes for lists where programmatic scrolling did not trigger scroll commands or macros
- Fix for stopping macros - commands waiting via delay property will now also be stopped
- Fix bug where javascript files with spaces in their name will fail when remote debugging is enabled
- Fix to list_join token on objects within a list
- Fix bug where CF.setProperties callback would not be called if duration was set to zero
- Fixed an issue where subpages would not reappear properly without valid transitions
- Fixed issue where subpage updates to off screen pages could trigger an assert.

Version 4.0.4

Date Released: NA

iViewer Next Date Released: Wednesday, 29th June 2011

iViewer Next Submitted to App Store: Wednesday, 22nd June 2011

Known Issues

- None

Changes

- Added CF.ObjectPressedEvent, CF.ObjectDraggedEvent and CF.ObjectReleasedEvent to allow more flexible JavaScript event handling.
- CF.listScroll now supports pixel-level scrolling.
- Tweaks for animations within lists via JavaScript.
- Improvements to remote monitor for system information, including showing list of clients connected to any TCP Server in iViewer.

- JavaScript calls in commands now work without requiring a JavaScript file added to the project.
- Buttons in “stretch mode” now also vertically stretch to full button height (and remain tiling horizontally).
- Overlay images can now be stretched to any size on the button.
- Added CF.getGuiDescription to JavaScript API [<http://www.commandfusion.com/docs/scripting>].
- Added CF.InputFieldEditedEvent to JavaScript API
- Changes to CF.request JavaScript method to increase flexibility and functionality including PUT requests.

Fixes

- Fixed bug with UDP broadcasting and multicasting.
- CF.setProperties() and CF.getProperties() now works for objects within list items.
- Fixes related to command sending from lists when scrolling tiny amounts.
- Fixed heartbeat disconnection issue while in background or resuming from sleep state.
- Fixed issue with lists and input fields not resizing via CF.setProperty calls.
- Fixed feedback processing order issue where onmatch items were always being processed before capture groups.
- Fix to ObjectPress/Drag/Release events.
- Fix for subpage resizing via JavaScript.
- Fixes for issues relating to CF.ConnectionStatusChangeEvent and the callback parameters.
- List scroll calls now work with only a single header or footer subpage in the list.
- Fixed bug where startup commands and macros for UDP systems were not being executed.
- Fixed cases where “push” transitions on pages would be reversed.
- Fixes to ensure CF.ip4address and similar members for network info are properly set.
- Fix for commands without a value (but containing JavaScript) being ignored.
- Fixes for timers on startup.
- Fixed issue where CF.setJoins wasn't matching documentation.
- Commands attached to sliders are now sent even if the slider has a zero join.
- Fixed issue with CF.listContents call returning invalid data.
- Fixed issue with input elements not properly sending join change events or updates to the control system.
- Fixes for string encodings.
- Fix for serial join change events not always being sent from input fields.

Version 4.0.3 build 157

Date Released: NA

iViewer Next Date Released: Thursday, 16th June 2011

iViewer Next Submitted to App Store: Saturday, 11th June 2011

Known Issues

- None

Changes

- Messages sent to offline TCP connections will no longer be queued by default. Instead they will be dropped by default. Queuing is now an option still via properties of the system.
- Updated code relating to HTTP requests to prevent rare crash.

Fixes

- Fixed issues with heartbeat and connection management for Control Systems and the warning dialog not hiding in GUI debug mode.
- Fixed issue where running on iOS 3.x devices would fail.

- Fixed issue where JavaScript files would not be run in an encrypted project archive output if using a non-default asset folder.
- Fixed bug where analog join changes from sliders were not triggering JoinChange events in JavaScript.
- Fixed issue where ipv4addresshex global token was containing the ipv4netmask.
- Fixes related to command sending from lists when scrolling tiny amounts.
- Join changes are now accurately sent from iViewer to Control System if the join is not attached to an object in the GUI.

Version 4.0.2

Date Released: NA

iViewer Next Date Released: Saturday, 4th June, 2011

iViewer Next Submitted to App Store: Tuesday, 31st May 2011

Known Issues

- CFsetProperties() does not work for objects within list items

Changes

- Debug monitor will not attempt auto reconnect after disconnection to avoid any startup issues. You must now manually refresh the debug logger after each iViewer launch
- Added support for automatically launching debugger from guiDesigner Upload Service
- Added support for sending “finished caching” update to guiDesigner to let user know when upload is complete
- Added support for continuous panning gesture (dragging one finger around)
- Added global tokens and JS constants for subnet masks: ipv4netmask and ipv6netmask, CF.ipv4netmask and CF.ipv6netmask

Fixes

- Origin and destination ports are now correctly used for UDP communication
- TCP Server now accepts connections from any IP when defining the IP address of the system as 0.0.0.0
- TCP Server now correctly only sends data once
- Fixed issue with animating subpages becoming partially non-responsive
- Stopped digital joins from going low on Control System connection/disconnection. It is now up to the Control System side to enforce join status changes on connection changes
- Fixed potential crash when clients connected to iViewer TCP Server on app exit
- Fixed issue where serial joins on buttons were not being updated until button was pressed
- Fixed issue where CF.listScroll() was not working due to current values not being up to date
- Fixed bug in CF.listAdd data processing

Version 4.0.1

Date Released: NA

iViewer Next Date Released: Friday, 27th May 2011

iViewer Next Submitted to App Store: Monday, 23th May 2011

Known Issues

- Modifying systems properties, connecting or disconnecting systems defined in the GUI not supported yet in the JavaScript API
- Gesture End events sometimes do not fire if you press the screen with another finger during the gesture (iOS 4.3.0 ONLY!)

- Animating subpages via JavaScript can result in sections of the subpage becoming non-responsive to user input
- TCP Server must define an IP Address other than 0.0.0.0
- TCP Server will send data twice
- Serial joins on buttons are not being updated until button is pressed

New Features

- Math expression errors now report to the web monitor
- CFlogObject JavaScript function implemented for easier object debugging

Fixes

- Fixed screen flicker on some page flips relating to video objects on the page
- Slider active state rendering
- Hostname resolving fixed
- Fixed issues with Control System received data showing random characters
- Fixed issue with Control System heartbeats not working correctly
- Fixed startup orientation issues
- Fixed issue with page not moving input fields into view when keyboard popup is displayed
- Fixed issue where list button in simulate mode could become active after dynamically adding list items during a scroll event

Version 4.0

Date Released: NA

iViewer Next Date Released: Wednesday, 4th May 2011

iViewer Next Submitted to App Store: Wednesday, 27th April 2011

Known Issues

- List button in simulate mode can become active after dynamically adding list items during a scroll event
- Setting binary data in joins from JavaScript code is not supported yet in this beta (although sending binary data to remote systems is supported)
- Modifying systems properties, connecting or disconnecting systems defined in the [GUI](#) not supported yet in the JavaScript [API](#)
- Changing from WiFi to WAN using hostname will not resolve without restarting the app
- Slider active state renders incorrectly if the slider is not sized to match the image
- Screen flickers black for a split second on page flip if the origin page has a video element
- Gesture End events sometimes do not fire if you press the screen with another finger during the gesture (iOS 4.3.0 ONLY!).

New Feature Highlights

- JavaScript authoring support* with rich iViewer [API](#) providing full [GUI](#) and network control, UI animations, HTTP requests, generic modules.
- Built-in remote debugging monitor* for live visualization of joins, tokens and systems states, JavaScript logging and debugging.
- Ability to execute JavaScript code in button actions, gestures, commands
- Video Window, supporting H.264*, MPEG4* and MJPEG formats, including local file playback, caching and full security options.
- Advanced networking options added, such as UDP multicast send/receive, HTTP systems, Bonjour browsing and publishing, SSL and TCP Server, IPv6 support

- Much improved performance for *loopback* systems
- Additional gesture properties available such as x and y positions
- Support for zipped GUI archives (with optionally encrypted GUI, scripts and assets)
- Page flips and subpages are enabled on unlicensed devices if only loopback systems are used in the project
- New global tokens with device IP addresses: `ipv4address`, `ipv4addresshex` and `ipv6address`
- Added a token to every element containing its join info (`d10`, `s100`, `a4`, etc): `join`
- Improved startup performance
- Numerous overall improvements in both performance and functionality

*Features marked with * require iOS 4.0 or later*

Fixes

- Image scaling now performs smoothing
- Images that fail to load will no longer leave a spinning loading indicator in it's place
- Fixed issue where dynamic images would stop refreshing after waking from sleep
- Fixed cases where button commands could continue firing on page flips
- Fixed issue where digital join values could resolve to values other than 0 or 1
- Cache loading progress bar now more accurately shows the end of caching

Version 3.4.3

Date Released: NA

iViewer Next Date Released: Saturday, 19th February 2011

iViewer Next Submitted to App Store: Friday, 11th February 2011

Known Issues

- None

Changes

- Dynamic images in lists updated to support caching
- Option to enable/disable caching of image objects
- Option to show/hide a downloading indicator for image objects
- Option to clear the image when loading a new image after a URL change (default is to keep existing image showing until new image is fully loaded)
- Images have a slight fade effect in after loading
- Preload images setting enabled by default on new install
- All images now stretch to fit the size of the GUI object (button, slider, etc). Only exception is 'stretchy' buttons that stretch via image border values.
- Added two new options to feedback processing of hex data: `Feedback Properties#Hex Data|Hex Byte` and `Feedback Properties#Hex Data|Hex String`.
- Added 'auto focus' option for input fields. This forces the input field to get focus and show the keyboard automatically when the page/subpage is shown that contains the input field.
- Changed the order of token replacement and join getter replacement to allow referencing tokens on other joins, such as list tokens from one list to another.
- Changed behaviour when memory warnings occur - now subpages hidden from screen are purged from memory and recreated when called for display again. Subpages with web views are not purged so as not to affect their current status.
- Changed onval and offval behaviour in feedback capture groups for digital join targets. Now if an offval is defined, the captured value must match it to set the digital join low.

Bug Fixes

- Images now load correctly from non-default asset folders
- On Release action for sliders now fire correctly always
- Fixed issue where global tokens would sometimes not accept a new value
- Pressing a button assigned to same digital join as an input field now causes the input field to send any attached command/macro when hiding the keyboard.
- Images with URL-encoded characters in the URL now load correctly.
- Fixed issue where keyboard would rotate incorrectly when on a page with a single orientation only.
- Fixed issue where negative numbers would always result in 00 when converted to hex.
- Fixed issue where lists with header items would not update correctly when a new row was inserted rather than appended.
- Fixed issue where images could disappear from memory in low-memory situations without iViewer noticing it.

Version 3.4.2

Date Released: Thursday, 18th November, 2010

Submitted to App Store: Wednesday, 10th November, 2010

Known Issues

- Loading dynamic images in lists still needs some tweaking.

Changes

- Added Remote Debugging|NSLogger support, allowing for full remote logging and debugging via a Mac running Snow Leopard.

Bug Fixes

- Fixes for printf statements in math expressions
- Fixes for hex byte capture and decimal conversion
- Fixed problem with status bar rotation
- Gesture transitions for subpage hiding fixed
- Web view loading default URL correctly now
- Gestures sending join updates to control systems correctly now
- Fixed problem where encrypted files would not load

Version 3.4.1

Date Released: Monday, 25th October, 2010

Submitted to App Store: Friday, 8th October, 2010

Known Issues

- Loading dynamic images in lists still needs some tweaking.

Changes

- Slider indicators fade only when inactive and active states have different sizes. Fade is also faster.
- Reverse filling sliders.
- Option to display caching progress bar when reloading GUI file.

- Apple Retina Displays | Support for 'Retina Displays' via @2x filenames.
- Default global tokens can be set to persist their data across launches.
- Assign a macro to system startup (alongside existing startup command).
- Generating hex bytes from math expressions.
- Ability to stop all macros running, or a specific macro, when sending a command.
- Ability to send commands/macros via feedback capture.
- Set page flip or subpage transitions options based on gesture actions.
- Math parser upgrades to include many new features (min/max, if/then/else, etc).
- Orientation is reported to control systems on startup now.
- Ability to set web pages to remain loaded when hidden, or to be stopped and refreshed when hidden then shown (ie. within a subpage or after page flip or rotation).
- The current page name is sent to Control Systems.
- The digital join of a page (if assigned) will be held high whilst that page is open.

Bug Fixes

- Gestures now send join updates to Control Systems.
- Button states after programmatic page flip corrected.
- Button states after page rotation corrected.
- Subpage transition fixes when multiple transitions occur at once.
- Stop sounds triggering on startup for disconnection/connection join states.
- Gestures now send join updates to control system.
- Page rotation animations now match for both directions.
- Crash relating to lists moving during page rotation fixed.
- Single character hex bytes (0-f) can now be captured and converted to decimal (previously only double character hex bytes would work in the range 00-0F).

Version 3.4.0

Date Released: Friday, 27th August, 2010

Submitted to App Store: Tuesday, 17th August, 2010

Known Issues

- Loading dynamic images in lists still needs some tweaking.

Changes

- Gestures | Gesture support
- Advanced Button Actions | Advanced button actions
- Embedded web view support
- Sending a command and macro from single button press
- Added printf support for feedback transforms
- Added decimal precision adjustment for feedback transforms
- Subpage always on top option
- Send device ID on connection via serial join 10002
- Send commands from lists when scrolling or reaching end
- Send commands from input fields on text change and finishing text input
- Report list position via Control | System protocol
- URL | Schema support
- Updates the regex parsing library to version 4.0 (<http://regexkit.sourceforge.net/RegexKitLite/#ReleaseInformation>)

Bug Fixes

- iPad loading screen orientation jolt fixed.
- Optional target value assignment in commands not effecting status bar
- Subpage z-order fixes
- GUI's smaller than the device resolution are now loaded into the center of the screen
- Page transitions fading by default fixed
- Cache corruption issues fixed
- Button active state rendering issues fixed (for transparent bg colors)
- Allow iPod music to continue playing when app is launched
- Horizontal lists now start at the left
- Lists remember their position after page flips
- Button overlay images in active state only now supported
- Use Math|Expressions in option command target value assignments
- Dramatic speed improvements in list loading
- Dynamic images on hidden subpages no longer keep refreshing in the background

Version 3.3.2

Date Released: Tuesday, 13th April, 2010

Submitted to App Store: Friday, 9th April, 2010

Known Issues

- Loading dynamic images in lists still needs some tweaking.
- iPad loading screen orientation jolt.
- Cannot format decimal precision for feedback transforms.

Bug Fixes

- Subpage transitions for 'none' and 'fade' fixed.
- dynamic image flickering on update from Control System

Version 3.3

Date Released: Friday, 2nd April, 2010

Submitted to App Store: Thursday, 25th March, 2010

Known Issues

- Loading dynamic images in lists still needs some tweaking.

Changes

- iPad Support!
- Added disconnection join status for all systems.
- Can now reference cached images via serial join for dynamic image changing
- Pressing 'Done' button on popup keyboard now sends the input field value on serial join
- Can now match hex values in digital join feedback 'on value'
- Added new feedback processing features, including list creation, multiple target assignments and math expressions.
- Added tokens, assigned during feedback parsing and used when sending dynamic commands and macros.

- Added timers for pages and subpages to send commands and macros.

Bug Fixes

- Sliders within lists now updated using the correct protocol.
- Connection state joins now updating for Control Systems and External Systems
- List button states clearing when list is cleared
- States of buttons without background images now updating correctly
- Serial values now correctly showing empty when blank serial join received via Control System
- Page flips via regex now working correctly.
- Subpage transitions tweaked.

Version 3.2

Date Released: Sunday, 10th January, 2010

Submitted to App Store: Thursday, 7th January, 2010

Known Issues

- Loading dynamic images in lists still a little buggy, working on fix for next release.
- Sliders within lists aren't being updated using the correct protocol. Working on fix for this release.

Changes

- Refresh feature added to images, allowing them to automatically refresh the image at a custom rate (in milliseconds)
- Subpages can now be toggled via button press without interaction from server (client side button toggling).

Bug Fixes

- Shadow rendering for all text elements (including buttons) now works correctly, with negative x/y offsets possible.
- Sending slider value in hex is now working correctly.

Version 3.1

Date Released: Tuesday, 7th December, 2009

Submitted to App Store: Thursday, 26th November, 2009

Changes

- Sending commands on slider press/drag/release now supported.
- Multiple GUI files can now be cached and switched between via button press, making multiple system connection profiles possible.
- Connection status of the main ControlSystem can be assigned to a digital join to show status within the GUI via button state for example.
- Feedback from systems with values in hex bytes or ascii representation of hex can now be parsed into numeric values for sliders/gauges/etc.
- Status bar now renders as semi-transparent to show it overlaying the GUI, rather than the default grey look.

Bug Fixes

- Connections via 3G will now automatically reconnect after device falls asleep.
- Parsing feedback that contains null bytes (x00) now works.

- Fixed rendering of overlay images with negative offset values, allowing overlay images to be moved over edges of buttons, etc.
- Fixed page orientations becoming out of sync. Now the correct orientation should always render. Had to sacrifice some animations on the page rotations for this (for now).

Version 3.0.2

Date Released: Thursday, 8th October, 2009

Submitted to App Store: Friday, 25th September, 2009

Known Issues

- Connections via 3G will not automatically reconnect after device falls asleep. Current workaround is to open Safari to re-enable 3G connections, then relaunch iViewer.
- Parsing feedback that contains null bytes (x00) does not work. Fix coming.
- Page flips on landscape pages can confuse the orientation and the page may become rendered sideways. Fix coming.

Changes

- Connection errors and warnings can now be suppressed for Control Systems by unchecking “Enable debug mode” in guiDesigner project properties.

Bug Fixes

- Fixed loading dynamic images in lists
- Fixed sending null hex character (x00) in commands
- Fixed sending startup commands to External Systems over UDP

Version 3.0.1

Date Released: September 2009

Known Issues

- Sending of the null hex character (x00) is not supported. Will be fixed in next release.
- External System | External Systems set to UDP do not send their assigned Startup Command on iViewer launch.
- Loading dynamic images in lists is limited to previously cached images, introduced this bug when fixing the dynamic image refresh flicker.

Changes

- Complete re-write of the communications code base.
- Added support for feedback processing (two way communications)
- Added iPhone iViewer Settings#Proximity Sensor | proximity sensor support
- Added iViewer Settings#Auto Lock Delay | auto-lock delay settings
- Added error message when trying to page flip or show subpages on an unregistered device
- Changed iViewer Settings | Settings to display legal information relating to third party libraries used in the code

Bug Fixes

- Fixed bug where GUI File cache would be lost after updating iViewer to a new version
- Fixed bug where changing a dynamic image would cause it to flicker (show no image before the newest image was loaded).

