

# Titanium SDK 7.5.0.GA Release Note

## Titanium SDK 7.5.0.GA - 15 November 2018

- [About this release](#)
- [Notice of feature and behavior Changes](#)
  - [Android platform](#)
  - [iOS platform](#)
  - [Windows platform](#)
- [New features](#)
  - [Android platform](#)
  - [iOS platform](#)
  - [Windows platform](#)
  - [Multiple platforms](#)
- [Community credits](#)
- [Fixed issues](#)
  - [Android](#)
  - [iOS](#)
  - [Windows](#)
  - [Multiple platforms](#)
- [Improvements](#)
  - [Android](#)
  - [iOS](#)
  - [Windows](#)
  - [Multiple platforms](#)
- [Known issues](#)
- [API changes](#)
  - [New APIs](#)
  - [Deprecated APIs](#)

## About this release

Titanium SDK 7.5.0.GA is a minor release of the SDK, addressing high-priority issues from previous releases.

Note: building of proxy bindings was improved and a validation step was added to ensure that no clashes occur with the combinations of using `@Kroll.method`, `@Kroll.getProperty`, and `@Kroll.setProperty` annotations with properties listed in `propertyAccessors`. The build script may fail with a message describing the class and will propose a solution to prevent it based on the type of clash.

We also addressed an issue on Windows that fixes a UI rendering performance issue that was causing the UI to freeze.

As of this release, Titanium SDK 7.4.x will not receive updates more than six months after the release of 7.5.0.GA (2019-05-15). Any needed fixes will be in 7.5.x or later supported releases within the 7.x branch. See [Axway Appcelerator Deprecation Policy](#) and [Nominal Lifetimes](#) documents for details.

## Notice of feature and behavior Changes

### Android platform

- [TIMOB-26258](#) - Android P: Implement Support
  - Added support for Android P including
    - Supporting inseting and display-cutout
    - Updated `Ti.Android.Service` to handle Android P `FOREGROUND_SERVICE` permission
    - Modified `AndroidManifest.xml` template to set `<application/>` attribute `android:usesCleartextTraffic` to `true` for backward compatibility (which can be set to `false` by modifying `tiapp.xml`)

### iOS platform

### Windows platform

- [TIMOB-25774](#) - Windows: Contact request should not block UI thread
  - Changed how access to contact using `ContactManager:RequestStoreAsync` no longer blocks UI thread due to `RequestStoreAsync`'s usage of the UI thread
- [TIMOB-25891](#) - Windows: Use of deprecated methods in `Ti.Media`
  - Windows 10 no longer requires `BackgroundMediaPlayer` in order to play audio in background (see [Play media in the](#)

background). It also doesn't require `MediaElement`, now `MediaPlayer` handles background audio when it is configured with new capability `<uap3:Capability Name="backgroundMediaPlayback" />`.

## New features

### Android platform

- **TIMOB-19618** - Android : add mixed content control to webview
  - Implemented `Ti.UI.WebView.mixedContentMode` which allows the loading of insecure resources from a secure origin

```
const win = Ti.UI.createWindow({ backgroundColor: 'gray' }),
  webView = Ti.UI.createWebView({
    url: 'https://www.bennish.net/mixed-content.html',
    mixedContentMode: true
  });

win.add(webView);
win.open();
```

- **TIMOB-24317** - Android: Extend `Ti.UI.View` "swipe" gesture properties using `ScaleGestureDetector`
  - Implemented focus area for view pinch event
- **TIMOB-24765** - Android O: Pinning Shortcuts and Widgets
  - Implemented `Ti.UI.ShortcutItem` pinning

```
var win = Ti.UI.createWindow({backgroundColor: 'grey'}),
  btn = Ti.UI.createButton({title: 'PIN SHORTCUT'}),
  shortcut = Ti.UI.createShortcutItem({
    id: 'test_shortcut',
    icon: Ti.Android.R.drawable.ic_dialog_info,
    title: 'TEST',
    description: 'DESCRIPTION'
  });

Ti.App.addEventListener('shortcutitemclick', function() {
  win.backgroundColor = 'blue';
});

btn.addEventListener('click', () => {
  win.backgroundColor = 'red';
  shortcut.pin();
});

win.add(btn);
win.open();
```

- **TIMOB-24817** - Android: `ListView` item template support CommonJS modules
  - Added `ListView` item template support to CommonJS modules
- **TIMOB-25635** - Android: Multi-View `ViewPager` (`ScrollableView`)
  - Added `clipViews` and `padding` properties to `ScrollableView`

> Expand

```
var win = Ti.UI.createWindow({
  title: 'ScrollableView clipViews',
  backgroundColor: '#eee'
});
// Common params
var params = {
  clipViews: false,
  showPagingControl: false,
  top: 0
};
if (Ti.Platform.name === 'android') {
  // Android specific params
  params.padding = {
    left: 40,
    right: 40
  };
} else {
  // iOS specific params
  params.width = '90%';
}
var scrollableView = Titanium.UI.createScrollableView(params);
var view1 = Ti.UI.createView({ id: 'view1', backgroundColor: '#836' });
var view2 = Ti.UI.createView({ id: 'view2', backgroundColor: '#246' });
var view3 = Ti.UI.createView({ id: 'view3', backgroundColor: '#48b' });
scrollableView.setViews([view1, view2, view3]);
win.add(scrollableView);
win.open();
```

source

- [TIMOB-25855](#) - Android: Add TextArea lines and maxLines support
  - Added support for TextArea lines and maxLines

> Expand

```
var win = Ti.UI.createWindow({ backgroundColor: '#fff' });
var sv = Ti.UI.createScrollView({
  layout: "vertical",
  height: Ti.UI.FILL
});
var tf1 = Ti.UI.createTextArea({
  borderColor: "#000",
  borderWidth: 1,
  color: "#000",
  width: 200,
  hintText: "1 to 5",
  hintTextColor: "#999",
  lines: 1,
  maxLines: 5
});
var tf2 = Ti.UI.createTextArea({
  lines: 2,
  maxLines: 2,
  borderColor: "#000",
  borderWidth: 1,
  color: "#000",
```

source

```

width: 200,
hintText:"2 to 2 -> setting lines to 4",
hintTextColor:"#999"
});
tf2.lines = 4;

var tf3 = Ti.UI.createTextArea({
  lines: 2,
  maxLines: 2,
  borderColor: "#000",
  borderWidth: 1,
  color: "#000",
  width: 200,
  hintText:"2 to 2",
  hintTextColor:"#999"
});
tf3.maxLines = 2;

var tf4 = Ti.UI.createTextArea({
  lines: 2,
  maxLines: 2,
  borderColor: "#000",
  borderWidth: 1,
  color: "#000",
  width: 200,
  hintText:"2 to 2 -> setting max to 4",
  hintTextColor:"#999"
});
tf4.maxLines = 4;

var textField1 = Ti.UI.createTextField({
  borderColor: "#000",
  borderWidth: 1,
  color: "#000",
  width: 200,
  hintText:"Textfield",
  hintTextColor:"#999"
});
var lbl1 = Ti.UI.createLabel({
  color:"#000",
  bottom: 10,
  text:"one line"
});
var lbl2 = Ti.UI.createLabel({
  color:"#000",
  bottom: 10,
  text:"two\nline"
});
var lbl3 = Ti.UI.createLabel({
  color:"#000",
  bottom: 10,
  text:"max 3 multi\n1 multi\n2 multi\n3 multi\n4 multi\nmulti\nline",
  maxLines: 3
});
var lbl4 = Ti.UI.createLabel({
  color:"#000",
  bottom: 10,
  text:"line 2 multi\n1 multi\n2 multi\n3 multi\n4 multi\nmulti\nline",
  lines: 2

```

```
});  
  
win.add(sv);  
sv.add(tf1);  
sv.add(tf2);  
sv.add(tf3);  
sv.add(tf4);  
sv.add(textField1);  
sv.add(lbl1);  
sv.add(lbl2);
```

```
sv.add(lbl3);
sv.add(lbl4);
win.open();
```

- [TIMOB-26427](#) - Android: Add "safeAreaPadding" property to `Ti.UI.Window`
  - Added support for `extendSafeArea`, `safeAreaPadding`, and `inset/notch` in `Ti.UI.Window`
- [TIMOB-26459](#) - Android: Add support for translucent status bar and navigation bar
  - Added support for translucent status bar and navigation bar
- [TIMOB-26484](#) - Android: Add "sound" support to `NotificationChannel`
  - Added sound support to `NotificationChannel`

## iOS platform

- [TIMOB-24776](#) - iOS: `ListView` item template support CommonJS modules
  - Added `ListView` item template support to CommonJS modules
- [TIMOB-24973](#) - iOS: Background audio-recording not working when screen is locked
  - Implemented feature that allows background audio recording when screen is locked

## Windows platform

- [TIMOB-19934](#) - Windows: Implement `Ti.Network.Cookie`
  - Implemented `Ti.Network.Cookie` for Windows
- [TIMOB-25868](#) - Windows: Implement `WebView.onlink` callback
  - Implemented `WebView.onlink` callback

```
var window = Ti.UI.createWindow(),
    webView = Ti.UI.createWebView({
      url: 'https://en.wikipedia.org/wiki/Portable_Document_Format',
      onlink: function (e) {
        if (e.url.endsWith('.pdf')) {
          alert('PDF: ' + e.url);
          return false;
        }
        return true;
      }
    });
window.add(webView);
window.open();
```

[Expand](#)

[source](#)

- [TIMOB-25881](#) - Windows: Implement `WebView.blacklistedURLs`
  - Implement `WebView.blacklistedURLs`

> Expand

```
var win = Titanium.UI.createWindow();
var webview = Titanium.UI.createWebView({
  blacklistedURLs: ['http://www.appcelerator.com/']
});
win.addEventListener('open', function () {
  webview.url = 'http://www.appcelerator.com/';
});
webview.addEventListener('blacklisturl', function (e) {
  alert('WebView blocked loading ' + e.url);
});
win.add(webview);
win.open();
```

source

- [TIMOB-26078](#) - Windows: Remove Ti.UI.currentWindow
  - Removed Ti.UI.currentWindow

## Multiple platforms

- [TIMOB-13379](#) - TiAPI: Add support for console.time and console.timeEnd
  - Implemented console.time and console.timeEnd according to <https://console.spec.whatwg.org/#time>
- [TIMOB-13966](#) - TiAPI: Select Locale for L() at runtime
  - Implemented feature that lets the user select which language to use at runtime

**est-Setup (requires a hello\_world language file in i18n/de and i18n/en)**

> Expand

```
var win = Ti.UI.createWindow({
  backgroundColor: '#fff'
});

var label = Ti.UI.createLabel({
  text: 'Current Locale: ' + Ti.Locale.currentLanguage + ' (' +
L('hello_world') + ')',
  top: 100
})

var btn = Ti.UI.createButton({
  title: 'Change to German'
});

btn.addEventListener('click', function() {
  Ti.Locale.setLanguage('de');
});

win.add(btn);
win.add(label);
win.open();
```

source

- [TIMOB-25048](#) - TiAPI: Android/iOS "ApplicationShortcuts" parity
  - Parity for moving ApplicationShortcuts to Ti.UI namespace
- [TIMOB-25985](#) - TiAPI: Disable accessibility features on Ti.UI.Button by using AttributedString
  - Added attributedString to Ti.UI.Button
- [TIMOB-26115](#) - TiAPI: Add support for console.timeLog

- Added support for `console.timeLog`

```

• console.time('mytimer'); // Start timer
  console.timeLog('mytimer'); // Log time taken so far
  console.timeLog('mytimer', 'with', 'some', 'extra', 'info'); // Log time
    taken with extra logging
  console.timeLog('mytimer', [ 'a', 'b', 'c' ], { objects: true }); // Should
    handle Arrays and Objects
  console.timeEnd('mytimer');

```

- Responds with

```

[INFO] mytimer: 1ms
[INFO] mytimer: 2ms with some extra info
[INFO] mytimer: 3ms ["a","b","c"] {"objects":true}
[INFO] mytimer: 3ms

```

- [TIMOB-26517](#) - CLI: Add link to implicit global warning log
  - Added doc link to global scope warning log

## Community credits

- Ezoom Agency - [TIMOB-13642](#)
- Michael Woode - [TIMOB-16511](#)
- Alejandro F. Carrera - [TIMOB-16678](#)
- Hazem Khaled - [TIMOB-24321](#)
- Richard Lustemberg - [TIMOB-24452](#), [TIMOB-26369](#)
- Varun Joshi - [TIMOB-25513](#)
- Uriel Lizama - [TIMOB-25391](#)
- Jorge Macias Garcia - [TIMOB-25756](#)
- Guillermo Figueras - [TIMOB-25810](#)
- Igor Stojanovic - [TIMOB-25901](#)
- Joe Falcone - [TIMOB-25971](#)
- Joshua A. Ceaser - [TIMOB-26047](#)
- Alexander de Greiff - [TIMOB-26104](#)
- Said Chaouche - [TIMOB-26127](#)
- David van de Meer - [TIMOB-26141](#), [TIMOB-26160](#), [TIMOB-26164](#), [TIMOB-26203](#)
- Russell Kenny - [TIMOB-26155](#), [TIMOB-26168](#)
- Josh Mocek - [TIMOB-26161](#)
- Gautier Drusch - [TIMOB-26182](#)
- Sergey Volkov - [TIMOB-26204](#), [TIMOB-26205](#), [TIMOB-26436](#)
- Zaedy Dantas Sayão - [TIMOB-26295](#)
- Michael Gangolf - [TIMOB-26314](#), [TIMOB-26374](#), [TIMOB-25635](#), [TIMOB-25855](#), [TIMOB-25043](#)
- Luc-Edmond Gaspard - [TIMOB-26317](#)
- Simon Buckingham - [TIMOB-26462](#)
- Mads Møller - [TIMOB-13966](#)
- Itay Szekely - [TIMOB-19618](#)
- Monili Nicolò - [TIMOB-25985](#)
- Omri Sapir - [TIMOB-26297](#)
- Flavio De Stefano - [TIMOB-24080](#)
- Andrea Vitale - [TIMOB-26073](#)
- Menno Juijtdit - [TIMOB-26174](#)
- Joren Vos - [TIMOB-26278](#), [TIMOB-26294](#)
- Prashant Saini - [TIMOB-26477](#)
- Thomas Neerup Andersen - [TIMOB-26479](#)

## Fixed issues

### Android

- [TIMOB-13642](#) - Android: Remove need for leading slash '/' to load images
- [TIMOB-16678](#) - Android: `File.read()` of JS files under Resources directory returns empty string for device/production builds



- TIMOB-23994 - Android: Using `__filename` in `app.js` errors with Can't find variable: `__filename`
- TIMOB-24321 - Android: Add `pivotX` and `pivotY` to `view`
- TIMOB-24605 - Android: Setting "backgroundImage" to a web URL causes a crash if JS is running on the UI thread
- TIMOB-25756 - Android: Restart app on system locale change
- TIMOB-25810 - Android: `ActionBar` height won't resize on screen rotation
- TIMOB-25932 - Android: Cannot build SDK with JDK 10 (missing `javah` tool)
- TIMOB-25979 - Android: Changing index break events on `TableRow`
- TIMOB-26030 - Android: `TCP read()` and `write()` methods cause a crash when running on main UI thread
- TIMOB-26079 - App does not run if included module have multiple `.aar` file.
- TIMOB-26080 - Android: `ERR_INVALID_CALLBACK` error when building a app/module on Node 10
- TIMOB-26104 - Android: Removing and re-adding drop-down picker resets last selection
- TIMOB-26124 - Android: `"Ti.buildHash"` property always returns null
- TIMOB-26127 - Android: Video Quality undefined
- TIMOB-26141 - Android: Accessing `Titanium.App.Android.R` logs GC error
- TIMOB-26149 - Android: `Ti.Filesystem.File#getDirectoryListing()` does not return null for file/non-existent directory
- TIMOB-26155 - Android: `ListView` `setItems` causes error after `updateItemAt` used with custom template
- TIMOB-26161 - Android: Activity Listening Broken on run-on-main-thread = true
- TIMOB-26168 - Android: `ScrollView` does not show `ScrollIndicators`
- TIMOB-26205 - Android: CLI does not provide useful information when a selected `buildTools` version is not installed
- TIMOB-26181 - Android: Application throws exception after close on 7.4.0
- TIMOB-26182 - An exception is raised when a remote image URL is set on an `ImageView` on some Android devices
- TIMOB-26203 - Android: Prevent exception in Log for HTTP response
- TIMOB-26204 - Android: `TransformAfterLayout` `NullPointerException` in `Ti.Animation`
- TIMOB-26205 - Android: `getActionBar()` throws "CalledFromWrongThreadException" exception
- TIMOB-26249 - Android: `ImageView` "image" set to a URL while "autorotate" is true will crash if running on main thread
- TIMOB-26253 - Android: `NotificationManager.notify()` will crash while screen is off and if app is missing `WAKE_LOCK` permission
- TIMOB-26274 - Android: Drawer hamburger menu disappears on click
- TIMOB-26284 - Android: Cannot add child-views to views extending `UIFragment`
- TIMOB-26295 - Android: Listview with template causes a nullpointer on `itemclick`
- TIMOB-26317 - Android: exception handler throws a `NullPointerException`
- TIMOB-26374 - Android: `Ti.UI.Button` can't be disabled until displayed as of 6.1.0
- TIMOB-26378 - Android: `backgroundDisabledColor` is ignored
- TIMOB-26442 - Android: `Toolbar` with "extendBackground" true is too tall if both status bar and navigation bar is transparent
- TIMOB-26451 - Android: `Hyperloop` does not generate meta data for synchronized classes
- TIMOB-26455 - Android: `Hyperloop` should sanitize numeric names
- TIMOB-26460 - Android: `Toolbar` height set to `Ti.UI.SIZE` won't resize when changing orientation
- TIMOB-26477 - `Ti.Locale.setLanguage` is not applying the locale effects properly on Android & iOS
- TIMOB-26496 - Android: Exception message not displaying for Java runtime exceptions
- TIMOB-25845 - Android: Non-production app builds with modules crash on startup if TLS 1.1 or higher is required for Internet access-
- TIMOB-26506 - Android: Validate `TiUIAbstract` `getWindowProxy()`
- TIMOB-26514 - Android: Custom theme applied to `ActionBar/Toolbar` title is lost when changing orientation as of 7.5.0.RC
- TIMOB-26516 - Android: `Ti.Platform` "id" and "version" properties return null if analytics disabled as of 7.5.0.RC
- TIMOB-26529 - Android: Amend `Ti.Platform` properties
- TIMOB-26534 - Android: Modify native HTTP handling to only cache "GET" responses
- TIMOB-26548 - Android: Unable to specify x86 as an abi in `tiapp` for production builds

## iOS

- TIMOB-16511 - iOS: Text in label gets truncated (...) when changing `fontWeight` from normal to bold
- TIMOB-23995 - iOS: Using `__dirname` in `app.js` errors with Can't find variable: `__dirname`
- TIMOB-24452 - iOS: Handling responses from `Ti.URLSession UploadTask` API in the SDK
- TIMOB-25391 - iOS: `TiUITable` crashes app with a Zombie
- TIMOB-25850 - iOS: `SearchBar` doesn't show `Bookmark` button when set on creation in `TableView`
- TIMOB-25866 - iOS : `TiUISearchBar` getter is returning value from proxy not from `searchbar` view
- TIMOB-25901 - iOS: App crashes when background services expire and queue tries to flush
- TIMOB-25942 - iOS 11: Screen orientation not working in navigation window
- TIMOB-25971 - iOS: `minimumFontSize` property in `Ti.UI.label` lowers text baseline as it decreases
- TIMOB-26032 - iOS/Android: `Ti.Contacts.getPersonByIdentifier` does not exist (Android), parity is incorrect
- TIMOB-26047 - iOS: Opening camera on iPhone X shows splash screen after permissions accepted
- TIMOB-26077 - iOS: `Ti.UI.convertUnits` returns incorrect values for 3x resolution devices
- TIMOB-26111 - iOS: Fix static analyzer warnings and possible leaks
- TIMOB-26121 - iOS: Default simulator should be compatible with selected xcode
- TIMOB-26160 - iOS: `ScrollableView + run-on-main-thread` - views dont render until scroll when re-adding views
- TIMOB-26164 - iOS: `TableView + run-on-main-thread` - `insertRowAfter` directly after `updateRow` causes error / race crash
- TIMOB-26165 - iOS: Requiring a JSON file that contains single quotes throws error
- TIMOB-26179 - iOS: Property check on proxy always returns true
- TIMOB-26184 - iOS: Timer functions can not be re-assigned
- TIMOB-26211 - iOS: Percentage based layout rounds floating widths/heights incorrectly
- TIMOB-26216 - iOS CLI: Incremental build does not remove unencrypted JS files when switching deploy type from "development" to "test"
- TIMOB-26217 - iOS: File methods `isFile()` and `isDirectory()` are missing for encrypted assets

- TIMOB-26299 - iOS: Ti.Media.AudioPlayer test-suite crashes in some situations
- TIMOB-26313 - iOS: accessibilityLabel doesn't get properly set on Ti.UI.tableView
- TIMOB-26334 - iOS: New AVPlayer-based Ti.Media.audioPlayer crashes on second setUrl() call
- TIMOB-26345 - iOS CLI: Upload to AppStore fails if app contains non-alphanumeric characters
- TIMOB-26347 - iOS: setRemoteBackup(flag) outputs an ERROR message when used on individual files.
- TIMOB-26369 - iOS: unhandledException Ti.App event no longer working.
- TIMOB-26372 - iOS: Assigning text value in change event causes duplicated input on TextView
- TIMOB-26389 - iOS: Application throws an error if there is no Ti.UI.Window reference
- TIMOB-26436 - iOS: VIdeoPlayer time in seconds
- TIMOB-26462 - iOS: Build throws unhelpful error when connecting to log server if localhost does not resolve
- TIMOB-26479 - TiUtils.m missing check for iPhone XR and iPhone XS Max images
- TIMOB-26513 - iOS: Push notifications crash the app when used with third-party module
- TIMOB-26533 - iOS: Ti.Media.Audio player without url set is crashing while registering for event listener
- TIMOB-26559 - iOS: App crash and missing events for local notifications

## Windows

- TIMOB-25183 - Windows: Default language resource warning
- TIMOB-25253 - Windows: backgroundGradient should have a default startPoint and endPoint
- TIMOB-25254 - Windows: backgroundGradient.colors should accept an array of just colour values
- TIMOB-25352 - Windows: Ti.Database.install should not prefix the specified path with the resourcesDirectory
- TIMOB-25513 - Windows: Ti.Utils.base64encode behaves different between Android/iOS and Windows
- TIMOB-25780 - Windows: Ti.Filesystem.File can't handle multilingual filename
- TIMOB-25783 - Windows: HttpClient.clearCookies freezes app
- TIMOB-25796 - Windows: Adding a hyperloop created UI element to a View multiple times throws an error
- TIMOB-25935 - Windows: Ti.Utils.base64decode should ignore line breaks
- TIMOB-25949 - Windows: Wrong layout with bottom/right and center
- TIMOB-25983 - Windows: Ti.UI.TextArea is not wrapped when there is long text, it just show in single line
- TIMOB-26014 - Windows: Titanium.Media.MEDIA\_TYPE\_VIDEO not working
- TIMOB-26158 - Windows: Build fails with ENOENT error when using transpile
- TIMOB-26177 - Windows: "focus" and "blur" events bubbles to parent views when it shouldn't
- TIMOB-26457 - Windows: Window.exitOnClose doesn't work when only one Window is opened
- TIMOB-26510 - Windows: "Runtime Exception: Layout cycle detected." when launching kitchensink-v2
- TIMOB-26525 - Windows: ScrollView does not expand with Ti.UI.SIZE
- TIMOB-26540 - Windows: When using html in webView application closes

## Multiple platforms

- TIMOB-26081 - CLI: Fix new Buffer deprecation warnings in build scripts
- TIMOB-26333 - Generate source maps for transpiled code
- TIMOB-26360 - Fix API documentation generation
- TIMOB-26403 - kitchensink-v2 broken on latest SDK build on master

## Improvements

### Android

- TIMOB-25533 - Android: Allow HttpClient to send request in onload
  - Fixed pending HTTP request status
- TIMOB-25899 - Android: Modify JSON.stringify(Error) to not return "{}"
  - Modified JS Error object to expose non-enumerable properties via JSON.stringify()
- TIMOB-26128 - Android: Allow asset directory names with a leading underscore to be included in the build
  - Implemented feature that allows asset directory names with a leading underscore to be included in the build
- TIMOB-26144 - Android: Improve File.getDirectoryListing() performance in Resources directory
  - Improved File.getDirectoryListing() performance in Resources directory
- TIMOB-26173 - Android: Add the Android Studio default SDK/NDK install paths to the Android lookup
  - Added default Android Studio SDK install paths to lookup
- TIMOB-26174 - Android: add setCancelable method on Ti.UI.AlertDialog
  - Added the buttonClickRequired property to Ti.UI.AlertDialog which controls whether the dialog can be cancelled by clicking the Android system back button. Defaults to true.
- TIMOB-26218 - Android: Resolve Veracode string equality warnings
  - Replaced string reference equality with value equality
- TIMOB-26246 - Android: Handle Android P insetting/display-cutout
  - Added Ti.UI.Window property extendSafeArea to render content beneath insets
  - Unlike iOS, this property is false by default on Android. This is because insets can be of any size and on side, making it more problematic.
- TIMOB-26260 - Android: Update "Ti.Android.Service" to handle "FOREGROUND\_SERVICE" permission
  - Updated Ti.Android.Service to handle Android P FOREGROUND\_SERVICE permission

- **TIMOB-26261** - Android: Compile with SDK version 28 (Android P)
  - Added support to compile with API Level 28 (Android P)
- **TIMOB-26278** - Android: Added Ti.App event "userinteraction"
  - Added the `userinteraction` event when `onUserInteraction` is triggered
- **TIMOB-26297** - Android: Resolve ImageView "images" property so that it'll work when targeting Android P
  - Fixed MD5-hashing for Android 9
- **TIMOB-26306** - Android: Support faster SDK builds with parallel execution '-j' on Windows
  - Implemented feature that allows for fast parallel execute -j8 NDK builds on Windows
  - Note: this is for the Titanium SDK itself, not for Titanium apps
- **TIMOB-26314** - Android: Add versionCode to tiapp.xml and use version in AndroidManifest.xml
  - Added `versionCode` to `tiapp.xml`
- **TIMOB-26376** - Android: Remove "commons-codec.jar" and replace with Java "MessageDigest"
  - Replaced deprecated Apache hashing class `DigestUtils` with `TiDigestUtils`
  - Replaced deprecated Apache HTTP classes with Android `HttpsURLConnection`
  - Added support for TLS 1.1 and higher
- **TIMOB-26377** - Android: Allow non-https requests to work when targeting Android P
  - Modified `AndroidManifest.xml` template to set `<application/>` attribute `android:usesCleartextTraffic` to `true` for backward compatibility (which can be set to `false` by modifying `tiapp.xml`)
- **TIMOB-26424** - Android: Ti.App lifecycle events are not fired
  - Implemented missing `Titanium.App` events

> Expand  
[source](#)

```
Ti.App.addEventListener('resume', () => {
    Ti.API.info('RESUME');
});

Ti.App.addEventListener('pause', () => {
    Ti.API.info('PAUSE');
});

Ti.App.addEventListener('resumed', () => {
    Ti.API.info('RESUMED');
});

Ti.App.addEventListener('paused', () => {
    Ti.API.info('PAUSED');
});
```

- **TIMOB-26435** - Android: Update gradle to support JDK 11
  - Updated to gradle 4.10.2
  - Changed "Failed to run dexter" warning to "Failed to run gradle"
- **TIMOB-26481** - Android: Target API Level 28 (Android P) by default
  - Target API Level 28 (Android P) by default

## iOS

- **TIMOB-24080** - iOS: Add paragraph styles to `AttributedString`
  - Added support for paragraph styles to `AttributedString`

> Expand

source

```
var win = Ti.UI.createWindow({
    backgroundColor: '#fff'
});
var label = Ti.UI.createLabel();
var str = "Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Praesent quis dolor consequat turpis pharetra laoreet vitae a sapien.
Aliquam erat volutpat. Nulla sit amet tellus sapien, a pulvinar metus.
Suspendisse risus elit, dictum id molestie ac, ullamcorper id dui. Vivamus
dapibus, eros nec sodales semper, sem dui semper eros, id condimentum quam
tellus at ligula. Ut justo magna, tempor et vestibulum id, tincidunt sit
amet quam. Maecenas dictum metus vel diam blandit facilisis. Ut tincidunt
nibh non ligula sagittis nec tincidunt augue vehicula. Suspendisse sem dui,
ornare in condimentum ut, convallis ut quam. Proin pharetra augue sed
tortor aliquam iaculis. Etiam non erat lectus. In ac metus massa, quis
dictum metus. Quisque faucibus quam non leo fringilla sit amet mattis
mauris dictum. Duis viverra ipsum blandit dolor congue sed adipiscing
tortor porta. Nullam malesuada felis ut dolor dignissim faucibus.";

label.attributedString = Ti.UI.createAttributedString({
    text: str,
    attributes: [{
        type: Ti.UI.ATTRIBUTE_PARAGRAPH_STYLE,
        value: {
            alignment: Ti.UI.TEXT_ALIGNMENT_JUSTIFY,
            minimumLineHeight: 3,
            headIndent: 5,
            lineSpacing: 5,
            lineBreakMode: Ti.UI.ATTRIBUTE_LINE_BREAK_BY_WORD_WRAPPING,
        },
        range: [0, str.length]
    }]
});

win.add(label);
win.open();
```

- [TIMOB-24909](#) - iOS: Move `Ti.Media.AudioPlayer` to `AVPlayer` internally, remove external library-dependency
  - Moved `Ti.Media.AudioPlayer` to `AVPlayer`
- [TIMOB-25871](#) - iOS: Add ability to hide back button on `NavigationWindow`
  - Added ability to hide back button on `NavigationWindow`

> Expand

source

```
var win2 = Titanium.UI.createWindow({
    backgroundColor: 'red',
    title: 'Red Window'
});

var win1 = Titanium.UI.iOS.createNavigationWindow({ window: win2 });

var win3 = Titanium.UI.createWindow({
    backgroundColor: 'blue',
    title: 'Blue Window'
});

var button = Titanium.UI.createButton({ title: 'Open Blue Window' });
button.addEventListener('click', function(){
    win1.openWindow(win3, {animated:true});
});

win2.add(button);
var button2 = Titanium.UI.createButton({ title: 'Open Gray Window(Back
Button Hidden)' });
button2.addEventListener('click', function(){
    win1.openWindow(win4, {animated:true});
});

win3.add(button2);
win1.open();

var win4 = Titanium.UI.createWindow({
    backgroundColor: 'gray',
    title: 'Gray Window',
    hidesBackButton: true
});

var button4 = Titanium.UI.createButton({ title: 'Close Gray Window' });
win4.add(button4);

button4.addEventListener('click', function(){
    win1.closeWindow(win4, {animated:false});
});
```

- [TIMOB-25879](#) - iOS: Create modules in "dist" directory (Parity with Android)
  - Parity: package iOS module zip in dist directory
- [TIMOB-26073](#) - iOS: Implement support for custom TextField backgroundColor in SearchBar
  - Implemented support for custom textField background in Titanium.UI.SearchBar
- [TIMOB-26196](#) - iOS: Move Ti.Media microphone related APIs to AVAudioRecorder and remove 3rd-party library
  - Removed SCListener and implemented AVAudioRecorder
- [TIMOB-26268](#) - iOS: Make RefreshControl on ScrollView available on iOS < 10
  - Added support for UIRefreshControl for UIScrollView on iOS < 10
- [TIMOB-26294](#) - iOS: Added Ti.App event "userinteraction" and property "trackUserInteraction"
  - Added the userinteraction event when onUserInteraction is triggered

## Windows

- [TIMOB-19932](#) - Windows: Implement Ti.Network properties & functions

- Implemented `Ti.Network` properties and functions for Windows including `Ti.Network.addHTTPCookie`, `Ti.Network.removeHTTPCookies`, and `Ti.Network.getHTTPCookies`
- **TIMOB-25043** - Windows: `Ti.Network.Cookie.isValid()` is missing
  - Implemented `Ti.Network.Cookie`
- **TIMOB-26210** - Windows: Compiler warnings `PRI249: 0xdef00520 - Invalid qualifier`
  - Suppressed "`PRI249: 0xdef00520 - Invalid qualifier`" warnings during build
- **TIMOB-26311** - Windows: leverage extended execution restricted capabilities
  - Fixed compile error when `rescap` Capabilities are set
- **TIMOB-26331** - Windows: Improve build/launch time via Titanium CLI
  - Improved build and launch time when using Titanium CLI (~40% improvement)
    - Reduced re-build time on device by stopping unnecessary clean build
    - Reduced development build (local/device) time by stopping unnecessary bundle package
    - Reduced build time by stopping unnecessary `cmake` invocation from `msbuild`
    - Reduced build time by stopping unnecessary compilation for C++ files

## Multiple platforms

- **TIMOB-25862** - Parity: Support async `Ti.UI.Webview.evalJS` call on Android & iOS
  - Added support for async `Ti.UI.Webview.evalJS`
- **TIMOB-26120** - Update npm package dependencies and reduce security vulnerabilities listed
  - Updated npm package dependencies
- **TIMOB-26308** - Reduce splash screen image sizes
  - Reduced the classic splash screen size
- **TIMOB-26515** - `Ti.UI.iOS.NavigationWindow` marked as deprecated, but Android missing (7.5.0+)
  - Removed deprecation notice for `Ti.UI.iOS.NavigationWindow`

## Known issues

- **TIMOB-26499** - iOS: Third-party modules cannot be referenced anymore in Alloy templates
  - This issue should be resolved by the GA release
- **MOD-2467** - `Ti.Map`: Exception thrown when targeting Android P
  - Following the [Android bug tracker](#), this is a known issue and the workaround is to add the following to the manifest: `<uses-library android:name="org.apache.http.legacy" android:required="false"/>`

## API changes

### New APIs

The following APIs are new or have expanded platform support in release 7.5.0.

API	Type	Notes
<code>Global.console.timeLog</code>	method	Log duration taken so far for an operation. (New API, supported on Android, iPhone and iPad.)
<code>Titanium.Android.Activity.userinteraction</code>	event	Called whenever a key, touch, or trackball event is dispatched to the activity. (New API, supported on Android.)
<code>Titanium.Android.NotificationChannel.getSound</code>	method	Gets the value of the <code>Titanium.Android.NotificationChannel.sound</code> property. (New API, supported on Android.)
<code>Titanium.Android.NotificationChannel.setSound</code>	method	Sets the value of the <code>Titanium.Android.NotificationChannel.sound</code> property. (New API, supported on Android.)
<code>Titanium.Android.NotificationChannel.sound</code>	property	A URL to the sound to play. (New API, supported on Android.)
<code>Titanium.App.Android.shortcutitemclick</code>	event	Fired when a <code>Titanium.UI.ShortcutItem</code> is clicked. (New API, supported on Android.)
<code>Titanium.App.shortcutitemclick</code>	event	Fired when a <code>Titanium.UI.ShortcutItem</code> is clicked. (New API, supported on Android, iPhone and iPad.)
<code>Titanium.App.trackUserInteraction</code>	property	Indicates whether or not the user interaction should be tracked. (New API, supported on iPhone and iPad.)

Titanium.App.userinteraction	event	Called whenever an interaction with the window occurred. To be used together with the Titanium.App.trackUserInteraction property. (New API, supported on Android, iPhone and iPad.)
Titanium.Locale.change	event	Fired when the device locale changes. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_BUFFERING	property	Audio data is being buffered from the network. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_INITIALIZED	property	Audio playback is being initialized. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_PAUSED	property	Playback is paused. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_PLAYING	property	Audio playback is active. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_STARTING	property	Audio playback is starting. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_STOPPED	property	Audio playback is stopped. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_STOPPING	property	Audio playback is stopping. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_WAITING_FOR_DATA	property	Player is waiting for audio data from the network. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AUDIO_STATE_WAITING_FOR_QUEUE	property	Player is waiting for audio data to fill the queue. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AudioPlayer.allowsExternalPlayback	property	Indicates whether the player allows switching to "external playback" mode. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.complete	event	Fired when the audio has finished playing. (Added support for iPhone and iPad.)
Titanium.Media.AudioPlayer.externalPlaybackActive	property	Indicates whether the player is currently playing video in "external playback" mode. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.getAllowsExternalPlayback	method	Gets the value of the Titanium.Media.AudioPlayer.allowsExternalPlayback property. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.getExternalPlaybackActive	method	Gets the value of the Titanium.Media.AudioPlayer.externalPlaybackActive property. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.getMuted	method	Gets the value of the Titanium.Media.AudioPlayer.muted property. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AudioPlayer.getRate	method	Gets the value of the Titanium.Media.AudioPlayer.rate property. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.metadata	event	Fired when the timed metadata was encountered most recently within the media as it plays. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.muted	property	Indicates whether or not audio output of the player is muted. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AudioPlayer.rate	property	Indicates the desired rate of playback; 0.0 means "paused", 1.0 indicates a desire to play at the natural rate of the current item. In addition, 2.0 would mean that the audio plays twice as fast. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.release	method	Stops buffering audio data and releases audio resources. (Added support for iPhone and iPad.)
Titanium.Media.AudioPlayer.restart	method	Restarts (stops and stars) audio playback. (New API, supported on Android, iPhone and iPad.)

Titanium.Media.AudioPlayer.seekToTime	method	Moves the playback cursor and invokes the specified block when the seek operation has either been completed or been interrupted. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.setAllowsExternalPlayback	method	Sets the value of the Titanium.Media.AudioPlayer.allowsExternalPlayback property. (New API, supported on iPhone and iPad.)
Titanium.Media.AudioPlayer.setMuted	method	Sets the value of the Titanium.Media.AudioPlayer.muted property. (New API, supported on Android, iPhone and iPad.)
Titanium.Media.AudioPlayer.setRate	method	Sets the value of the Titanium.Media.AudioPlayer.rate property. (New API, supported on iPhone and iPad.)
Titanium.Media.QUALITY_640x480	property	Media type constant for medium-quality video recording. (Added support for Android.)
Titanium.Media.QUALITY_HIGH	property	Media type constant for high-quality video recording. (Added support for Android.)
Titanium.Media.QUALITY_IFRAME_1280x720	property	Media type constant for medium-quality video recording. (Added support for Android.)
Titanium.Media.QUALITY_LOW	property	Media type constant for low-quality video recording. (Added support for Android.)
Titanium.Media.getQUALITY_640x480	method	Gets the value of the Titanium.Media.QUALITY_640x480 property. (Added support for Android.)
Titanium.Media.getQUALITY_IFRAME_1280x720	method	Gets the value of the Titanium.Media.QUALITY_IFRAME_1280x720 property. (Added support for Android.)
Titanium.UI.ATTRIBUTE_PARAGRAPH_STYLE	property	Use with Attribute.type to manages the behaviour of string set. (New API, supported on iPhone and iPad.)
Titanium.UI.ActivityIndicator.anchorPoint	property	Coordinate of the view about which to pivot an animation. (Added support for Android.)
Titanium.UI.ActivityIndicator.getAnchorPoint	method	Gets the value of the Titanium.UI.ActivityIndicator.anchorPoint property. (Added support for Android.)
Titanium.UI.ActivityIndicator.setAnchorPoint	method	Sets the value of the Titanium.UI.ActivityIndicator.anchorPoint property. (Added support for Android.)
Titanium.UI.AlertDialog.anchorPoint	property	Coordinate of the view about which to pivot an animation. (Added support for Android.)
Titanium.UI.AlertDialog.buttonClickRequired	property	Setting this to true requires the end-user to click a dialog button to close the dialog. (New API, supported on Android.)
Titanium.UI.AlertDialog.getAnchorPoint	method	Gets the value of the Titanium.UI.AlertDialog.anchorPoint property. (Added support for Android.)
Titanium.UI.AlertDialog.getButtonClickRequired	method	Gets the value of the Titanium.UI.AlertDialog.buttonClickRequired property. (New API, supported on Android.)
Titanium.UI.AlertDialog.setAnchorPoint	method	Sets the value of the Titanium.UI.AlertDialog.anchorPoint property. (Added support for Android.)
Titanium.UI.AlertDialog.setButtonClickRequired	method	Sets the value of the Titanium.UI.AlertDialog.buttonClickRequired property. (New API, supported on Android.)
Titanium.UI.Android.FLAG_TRANSLUCENT_NAVIGATION	property	Window flag which makes the Android system's navigation bar semi-transparent. (New API, supported on Android.)
Titanium.UI.Android.FLAG_TRANSLUCENT_STATUS	property	Window flag which makes the Android system's top status bar semi-transparent. (New API, supported on Android.)
Titanium.UI.Android.ProgressIndicator.anchorPoint	property	Coordinate of the view about which to pivot an animation. (New API, supported on Android.)



Titanium.UI.Android.ProgressIndicator.getAnchorPoint	method	Gets the value of the Titanium.UI.Android.ProgressIndicator.anchorPoint property. (New API, supported on Android.)
Titanium.UI.Android.ProgressIndicator.setAnchorPoint	method	Sets the value of the Titanium.UI.Android.ProgressIndicator.anchorPoint property. (New API, supported on Android.)
Titanium.UI.Button.attributedString	property	Specify an attributed string for the label. (New API, supported on Android, iPhone and iPad.)
Titanium.UI.Button.getAttributedString	method	Gets the value of the Titanium.UI.Button.attributedString property. (New API, supported on Android, iPhone and iPad.)
Titanium.UI.Button.setAttributedString	method	Sets the value of the Titanium.UI.Button.attributedString property. (New API, supported on Android, iPhone and iPad.)
Titanium.UI.EmailDialog.anchorPoint	property	Coordinate of the view about which to pivot an animation. (Added support for Android.)
Titanium.UI.EmailDialog.getAnchorPoint	method	Gets the value of the Titanium.UI.EmailDialog.anchorPoint property. (Added support for Android.)
Titanium.UI.EmailDialog.setAnchorPoint	method	Sets the value of the Titanium.UI.EmailDialog.anchorPoint property. (Added support for Android.)
Titanium.UI.OptionDialog.anchorPoint	property	Coordinate of the view about which to pivot an animation. (Added support for Android.)
Titanium.UI.OptionDialog.getAnchorPoint	method	Gets the value of the Titanium.UI.OptionDialog.anchorPoint property. (Added support for Android.)
Titanium.UI.OptionDialog.setAnchorPoint	method	Sets the value of the Titanium.UI.OptionDialog.anchorPoint property. (Added support for Android.)
Titanium.UI.Picker.anchorPoint	property	Coordinate of the view about which to pivot an animation. (Added support for Android.)
Titanium.UI.Picker.getAnchorPoint	method	Gets the value of the Titanium.UI.Picker.anchorPoint property. (Added support for Android.)
Titanium.UI.Picker.setAnchorPoint	method	Sets the value of the Titanium.UI.Picker.anchorPoint property. (Added support for Android.)
Titanium.UI.ScrollableView.clipViews	property	Determines whether the previous and next pages are clipped, so that they are not visible adjacent to the current page. (Added support for Android.)
Titanium.UI.ScrollableView.getClipViews	method	Gets the value of the Titanium.UI.ScrollableView.clipViews property. (Added support for Android.)
Titanium.UI.ScrollableView.getPadding	method	Gets the value of the Titanium.UI.ScrollableView.padding property. (New API, supported on Android.)
Titanium.UI.ScrollableView.padding	property	The padding applied to the scrollable view. (New API, supported on Android.)
Titanium.UI.ScrollableView.setClipViews	method	Sets the value of the Titanium.UI.ScrollableView.clipViews property. (Added support for Android.)
Titanium.UI.ScrollableView.setPadding	method	Sets the value of the Titanium.UI.ScrollableView.padding property. (New API, supported on Android.)
Titanium.UI.SearchBar.fieldBackgroundDisabledImage	property	Background image of the text field in disabled state, specified as a local file path or URL. (New API, supported on iPhone and iPad.)
Titanium.UI.SearchBar.fieldBackgroundImage	property	Background image of the text field, specified as a local file path or URL. (New API, supported on iPhone and iPad.)
Titanium.UI.SearchBar.getFieldBackgroundDisabledImage	method	Gets the value of the Titanium.UI.SearchBar.fieldBackgroundDisabledImage property. (New API, supported on iPhone and iPad.)

Titanium.UI.SearchBar.getFieldBackgroundImage	method	Gets the value of the Titanium.UI.SearchBar.fieldBackgroundImage property. (New API, supported on iPhone and iPad.)
Titanium.UI.SearchBar.setFieldBackgroundDisabledImage	method	Sets the value of the Titanium.UI.SearchBar.fieldBackgroundDisabledImage property. (New API, supported on iPhone and iPad.)
Titanium.UI.SearchBar.setFieldBackgroundImage	method	Sets the value of the Titanium.UI.SearchBar.fieldBackgroundImage property. (New API, supported on iPhone and iPad.)
Titanium.UI.ShortcutItem	object	An application shortcut. (New API, supported on Android, iPhone and iPad.)
Titanium.UI.Tab.anchorPoint	property	Coordinate of the view about which to pivot an animation. (Added support for Android.)
Titanium.UI.Tab.getAnchorPoint	method	Gets the value of the Titanium.UI.Tab.anchorPoint property. (Added support for Android.)
Titanium.UI.Tab.setAnchorPoint	method	Sets the value of the Titanium.UI.Tab.anchorPoint property. (Added support for Android.)
Titanium.UI.TabGroup.anchorPoint	property	Coordinate of the view about which to pivot an animation. (Added support for Android.)
Titanium.UI.TabGroup.getAnchorPoint	method	Gets the value of the Titanium.UI.TabGroup.anchorPoint property. (Added support for Android.)
Titanium.UI.TabGroup.setAnchorPoint	method	Sets the value of the Titanium.UI.TabGroup.anchorPoint property. (Added support for Android.)
Titanium.UI.TextArea.getLines	method	Gets the value of the Titanium.UI.TextArea.lines property. (New API, supported on Android.)
Titanium.UI.TextArea.getMaxLines	method	Gets the value of the Titanium.UI.TextArea.maxLines property. (New API, supported on Android.)
Titanium.UI.TextArea.lines	property	Number of lines tall the text area height will be, if set. (New API, supported on Android.)
Titanium.UI.TextArea.maxLines	property	Maximum line count of text field input. (New API, supported on Android.)
Titanium.UI.TextArea.setLines	method	Sets the value of the Titanium.UI.TextArea.lines property. (New API, supported on Android.)
Titanium.UI.TextArea.setMaxLines	method	Sets the value of the Titanium.UI.TextArea.maxLines property. (New API, supported on Android.)
Titanium.UI.TextField.getPasswordRules	method	Gets the value of the Titanium.UI.TextField.passwordRules property. (New API, supported on iPhone and iPad.)
Titanium.UI.TextField.passwordRules	property	Set password rules that should be used for this text field. (New API, supported on iPhone and iPad.)
Titanium.UI.TextField.setPasswordRules	method	Sets the value of the Titanium.UI.TextField.passwordRules property. (New API, supported on iPhone and iPad.)
Titanium.UI.View.anchorPoint	property	Coordinate of the view about which to pivot an animation. (Added support for Android.)
Titanium.UI.View.getAnchorPoint	method	Gets the value of the Titanium.UI.View.anchorPoint property. (Added support for Android.)
Titanium.UI.View.setAnchorPoint	method	Sets the value of the Titanium.UI.View.anchorPoint property. (Added support for Android.)
Titanium.UI.WebView.getMixedContentMode	method	Gets the value of the Titanium.UI.WebView.mixedContentMode property. (New API, supported on Android.)
Titanium.UI.WebView.getOnlink	method	Gets the value of the Titanium.UI.WebView.onlink property. (New API, supported on Android, iPhone and iPad.)

Titanium.UI.WebView.mixedContentMode	property	If <code>true</code> , allows the loading of insecure resources from a secure origin. (New API, supported on Android.)
Titanium.UI.WebView.onlink	property	Fired before navigating to a link. (New API, supported on Android, iPhone and iPad.)
Titanium.UI.WebView.setMixedContentMode	method	Sets the value of the Titanium.UI.WebView.mixedContentMode property. (New API, supported on Android.)
Titanium.UI.WebView.setOnlink	method	Sets the value of the Titanium.UI.WebView.onlink property. (New API, supported on Android, iPhone and iPad.)
Titanium.UI.Window.extendSafeArea	property	Specifies whether the screen insets/notches are allowed to overlap the window's content or not. (Added support for Android.)
Titanium.UI.Window.getExtendSafeArea	method	Gets the value of the Titanium.UI.Window.extendSafeArea property. (Added support for Android.)
Titanium.UI.Window.getHidesBackButton	method	Gets the value of the Titanium.UI.Window.hidesBackButton property. (New API, supported on iPhone and iPad.)
Titanium.UI.Window.getSafeAreaPadding	method	Gets the value of the Titanium.UI.Window.safeAreaPadding property. (New API, supported on Android.)
Titanium.UI.Window.hidesBackButton	property	Set this to true to hide the back button of navigation bar. (New API, supported on iPhone and iPad.)
Titanium.UI.Window.safeAreaPadding	property	The padding needed to safely display content without it being overlapped by the screen insets and notches. (New API, supported on Android.)
Titanium.UI.Window.setExtendSafeArea	method	Sets the value of the Titanium.UI.Window.extendSafeArea property. (Added support for Android.)
Titanium.UI.Window.setHidesBackButton	method	Sets the value of the Titanium.UI.Window.hidesBackButton property. (New API, supported on iPhone and iPad.)
Titanium.UI.createShortcutItem	method	Creates and returns an instance of Titanium.UI.ShortcutItem. (New API, supported on Android, iPhone and iPad.)

## Deprecated APIs

The following APIs are deprecated in release 7.5.0. [Deprecated APIs](#)

API	Type	Notes
Titanium.Media.AudioPlayer.STATE_BUFFERING	property	Use Titanium.Media.AUDIO_STATE_BUFFERING instead.
Titanium.Media.AudioPlayer.STATE_INITIALIZED	property	Use Titanium.Media.AUDIO_STATE_INITIALIZED instead.
Titanium.Media.AudioPlayer.STATE_PAUSED	property	Use Titanium.Media.AUDIO_STATE_PAUSED instead.
Titanium.Media.AudioPlayer.STATE_PLAYING	property	Use Titanium.Media.AUDIO_STATE_PLAYING instead.
Titanium.Media.AudioPlayer.STATE_STARTING	property	Use Titanium.Media.AUDIO_STATE_STARTING instead.
Titanium.Media.AudioPlayer.STATE_STOPPED	property	Use Titanium.Media.AUDIO_STATE_STOPPED instead.
Titanium.Media.AudioPlayer.STATE_STOPPING	property	Use Titanium.Media.AUDIO_STATE_STOPPING instead.
Titanium.Media.AudioPlayer.STATE_WAITING_FOR_DATA	property	Use Titanium.Media.AUDIO_STATE_WAITING_FOR_DATA instead.
Titanium.Media.AudioPlayer.STATE_WAITING_FOR_QUEUE	property	Use Titanium.Media.AUDIO_STATE_WAITING_FOR_QUEUE instead.
Titanium.Media.AudioPlayer.play	method	Use the cross-platform API start instead.
Titanium.Media.AudioPlayer.setPaused	method	Use the cross-platform API pause instead.

Titanium.UI.ATTRIBUTE_LINE_BREAK	property	Use Titanium.UI.ATTRIBUTE_PARAGRAPH_STYLE.lineBreakMode instead.
----------------------------------	----------	--