

webMethods Mobile Suite 10.1 Readme

October 2017

This file contains important information you must read before using webMethods Mobile Suite 10.1. You can find system requirements, user documentation, and installation and upgrade instructions on the [Documentation website](#) or the [TECHcommunity website](#). At those locations, you can also find suite-related security and globalization information.

Included in this file is information about functionality that has been added, removed, deprecated, or changed for this product. Deprecated functionality continues to work and is supported by Software AG, but may be removed in a future release. Software AG recommends against using deprecated functionality in new projects.

1.0	Critical Information.....	1
2.0	Known Issues.....	2
3.0	Usage Notes.....	3
4.0	Fixes Included in Each Release.....	5
5.0	Other Resolved Issues.....	5
6.0	Documentation Changes	5
7.0	Added, Removed, Deprecated, or Changed Parameters.....	6
8.0	Added, Removed, Deprecated, or Changed APIs	6
9.0	Copyright Information.....	10
10.0	Support.....	10

1.0 Critical Information

This section lists any critical issues for the current release that were known when this readme was published. For critical information found later, go to the Knowledge Center on the [Empower website](#).

Mobile Administrator

MOBILE-3381

When running an iOS build job on a Mobile Administrator 9.12 or 10.1 build node using Xcode 8.3+ the build job fails with the following error message:

```
> xcrun: error: unable to find utility "PackageApplication", not a developer tool or in PATH
```

This error is related to Xcode 8.3+ coming without the *PackageApplication* script used in older versions

of Xcode for packaging application. Mobile Administrator requires Xcode 6 for building and packaging applications as of 9.12 and 10.1. If there is a reason for using Xcode 8.3+ instead of using the supported Xcode 6 version, please contact support for a workaround.

MOBILE-2586

Codesign fails when packaging an iOS app for an App Store build using Mobile Administrator. When doing App Store release builds on Mobile Administrator with build nodes running Mac OS X 10.10+, the build job might fail with the following error message:

```
> /usr/bin/codesign --force --preserve-metadata=identifier,entitlements,resource-rules --sign [...]  
--resource-rules=/var/folders/.../.app/ResourceRules.plist --entitlements [...]  
failed with error 1. Output: Warning: usage of --preserve-metadata with option "resource-rules"  
(deprecated in Mac OS X >= 10.10)!
```

This is related to a bug in Xcode command line tools used to build and package the app. There are two known workarounds to address this issue. First, for App Store builds you can use Multibuild locally, second is to remove the "`--resource-rules=$destApp/ResourceRules.plist`" argument for `codesign`, in `/.../Xcode.app/Contents/Developer/Platforms/iPhoneOS.platform/Developer/usr/bin/PackageApplication`.

Please contact support in case you experience any related problems.

2.0 Known Issues

This section lists any issues for the current release that were known when this readme was published. For known issues found later, go to the Knowledge Center on the [Empower website](#).

Mobile Designer

- MOBILE-3316
Opening a PDF file directly in `nUIWebView` by just providing local or remote URL does not open the preview of the PDF. As a workaround native code injection must be used. Fix will be provided.
- MOBILE-3367
Reusing `nUINavButtons` does not work, i.e. removing and re-adding the same instance to a view will not work properly.
- MOBILE-3366
Any UI changes triggered in the `onTabSelected()` callback won't be reflected in UI on Android. As a workaround explicitly call `syncUI()` at the end of `onTabSelected()`.

Mobile Development

- MOBILE-3358 `nUIListElement` not yet supported in Mobile Development
When using `ListViewElement` based templates in Mobile Development, generated code will not use newly introduced `nUIListElement`. Instead, `nUIContainer` with `nUITableButtons` based code will be generated like in previous versions of Mobile Suite.

3.0 Usage Notes

This section provides any additional information you need to work with the current release of this product.

- With Mobile Designer 10.1 an important change to the application runtime was made that all developers need to be aware of. The new runtime does not use the Mobile Designer Midlet Thread that was used in all versions up to 9.12 for synchronizing UI updates in the application. Instead it now utilizes the main thread provided by the particular native platform for all UI operations and synchronization. All operations now run on the (native) main thread. This is important for performance of the application and user experience. As in native applications, all heavy operations now need to run on new threads to not block the UI during execution. A set of new APIs (`runAsync()` and `ICallable`) was introduced to make it easy to execute operations in new threads and synchronizing back to the main thread.

It is also important to note that in case the UI is not updated automatically, running `syncUI()` on `nUIViewDisplay` will synchronize all changes and update the UI.

Please read the documentation for more information.

- When iOS App Store archive validation fails with the following error message:
`> This bundle is invalid. The info.plist contains an invalid key 'CFBundleResourceSpecification' in app bundle`
use `<property name="ios.use.resourcerules.plist" value="true"/>` in `targets/_default_.xml` to make Mobile Designer not add the `'CFBundleResourceSpecification'` key to generated `Info.plist`. If there are still problems when uploading iOS builds to Apple App Store, contact your support.
- With `nUIListView` and `nUIListElement` a new type of control (or better: a new type of view) was introduced. A ListView allows you to efficiently manage an endless number of any kind of `nUIDisplayObjects` in a single column. ListViews load and render only the `nUIDisplayObjects` visible on screen. As the user scrolls the ListView vertically, the managed `nUIDisplayObjects` are (lazy) loaded and rendered to provide best performance and user experience. This first version of `nUIListView` and `nUIListViewElements` supports pull-to-refresh and endless scrolling events to allow loading of data on demand.
- With `nUISwitch`, an alternative to `nUICheckbox` was introduced, that provides a better native experience for on and off states in your applications.
- An all new visual outline preview was introduced in Mobile Development Outline Editor, giving the developer instant visual feedback of the view or template being modeled. This is the first step towards real “what you see is what you get”-like modeling experience. The new Outline comes with all new property views to change colors, alignments, etc. The new outline preview will be available without further configuration and is available via the Outline view in Software AG Designer. On selection of preview-enabled objects (such as views, templates) in the AML Editor, the outline preview will automatically render a visual representation of the object.
- In Mobile Development, new icon and image asset editors were introduced. Picking the correct image assets for devices can be quite a tricky task for developers. Image sizes may depend on

screen resolution, pixel density, and device rotation, as well as any number of application-specific concerns. Most devices have a concept of an "Asset Catalog" - that is, multiple Images targeting the possible resolution/density setups are registered with one common name, and the Mobile OS selects the correct file to return for that name based on the information it has about the display. With the new icon and image asset editors and the new AssetImageLoader API, images and icons can be easily created and integrated into your product. At runtime the correct resolution will be used automatically.

- Using the Software AG Designer Example Wizard, a new sample app "Cocktails" can be added to your workspace. It shows loading and parsing of JSON data using EML modeling, feeding a data source and providing the data into the new *nUIListView* without coding. It also features a navigation drawer and shows how to use the new image asset and icon editors and how to use styling to define global UI properties. The whole app comes with almost zero custom code.
- A new File Management API was introduced. It provides full read and write access to files and folders on the filesystem of the device at runtime. To access files across all platforms, virtual root folders for documents, cache, resources and external storage are provided by the File Management API.
- With *RuntimeInfo* and *SystemInfo* classes, a new API was introduced to provide easy access to device specific runtime properties, such as screen dimension, ppi value, form factor, location enabled status, system info, etc.
- A new Connection API was introduced to simplify working with HTTP connections. The new API supports synchronous and asynchronous connections and makes the *HttpConnectionHandler* obsolete. To execute connections asynchronously in a new thread, use *Connection.executeAsync()*.
- For monitoring and polling the connectivity state of the device at runtime, a new API was introduced. It allows users to receive notifications when the device's network adapters change state and to find out about the connection type, such as Wifi, cellular, etc. Using *ConnectivityStatusRequester* and *IConnectivityWatcher* the application will get notifications on device connection state change without the need to poll the state.
- A new interface *IPushNotificationListener* was introduced for processing push notifications. This listener can be registered via *Application.setPushNotificationListener(IPushNotificationListener listener)*. Using *Application.onStartAppFromPushNotification(PushNotification message)*, it is now possible to find out if the application was started from a push notification.
- Phoney now has an option to start with the screen rotated. Passing "-start_landscape" or "-sl" will cause Phoney to rotate through 90 degrees before the application code is launched. Phoney also supports mouse-wheel in the new release.
- *nUIDisplayObject* now supports changing visibility using *setVisible()* API.
- With the introduction of OperationQueues, it is now possible to manage instances of *IOperation* more conveniently. Using queues it is easy to run operations in parallel or sequentially, pause and resume. Queues even help to execute operations when online or offline.

4.0 Fixes Included in Each Release

This section lists the latest fix level that has been included in each release for each product component. A release is listed in this section only if changes occurred in that release. Go to the Knowledge Center on the [Empower website](#) for detailed information about fixes.

Mobile Designer

Release 9.12

- WWW_9.12_MobileDesigner_Fix3

Mobile Development

Release 9.12

- WWG_9.12_MobileDevelopment_Fix3

5.0 Other Resolved Issues

This section lists the issues that were resolved in each release but were not part of the fixes listed in the previous section. A release is listed in this section only if changes occurred in that release.

6.0 Documentation Changes

This section describes significant changes to the documentation, such as the addition, relocation, or removal of product guides, online help, chapters, or other major content. A release is listed in this section only if changes occurred in that release.

Mobile Designer

Release 10.1

- The documentation was updated.

Mobile Development

Release 10.1

- The documentation was updated.

7.0 Added, Removed, Deprecated, or Changed Parameters

A release is listed in this section only if changes occurred in that release.

Release 10.1

Mobile Designer

Added Parameter	Description
android.manifest.permissions.exclude.SEND_SMS	To remove SMS_SEND permission from AndroidManifest.
ios.pn.device_token.style	Use via <i>RuntimeInfo.setProperty()</i> with value "raw" to get unformatted push tokens.
project.runtime.uses.nativeui.phoney	Enable to use new runtime classes.
Deprecated Parameter	Replacement, if any
project.handset.output.preedit.dir.root	
project.handset.output.postedit.dir.root.default	

8.0 Added, Removed, Deprecated, or Changed APIs

A release is listed in this section only if changes occurred in that release.

Release 10.1

Added API	Description
com.softwareag.mobile.reshandler.AnfTaskResourceHandler public void addDirContentAsAssetCatalogs(String dir)	
com.softwareag.mobile.runtime.nui.nUIDisplayObject public boolean isVisible() public void setVisible(boolean visible)	Hide / show <i>nUIDisplayObjects</i> at runtime.

Added API	Description
com.softwareag.mobile.runtime.nui.nUIWindowDisplay public IGlobalPropertiesIOS getIOSGlobalProperties() public IGlobalPropertiesAndroid getAndroidGlobalProperties() com.softwareag.mobile.runtime.nui.IGlobalPropertiesIOS com.softwareag.mobile.runtime.nui.IGlobalPropertiesAndroid	Access to iOS and Android specific properties as for example tint colors of <i>nUINavButtons</i> .
com.softwareag.mobile.runtime.core.Application public void onStartAppFromPushNotification(PushNotification message) public PushNotification getStartPushNotification() public IPushNotificationListener setPushNotificationListener(IPushNotificationListener listener) com.softwareag.mobile.runtime.pn.PushNotification com.softwareag.mobile.runtime.pn.IPushNotificationListener	Push Notifications
com.softwareag.mobile.md.digest.MD5Sum public String getDigest(String s) public String getDigest(String s, String encoding) public String getDigest(byte[] bytes)	MD5 Checksum support for <i>String</i> and <i>byte[]</i> types.
com.softwareag.mobile.core.MDApplication com.softwareag.mobile.core.IMDApplication com.softwareag.mobile.core.IMDApplicationListener com.softwareag.mobile.md.util.concurrent.IResultCallback	New entry point of every application.
com.softwareag.mobile.runtime.comms.Connection com.softwareag.mobile.runtime.comms.Response com.softwareag.mobile.runtime.comms.HttpResponse com.softwareag.mobile.runtime.comms.IConnectionCallback	New Connection API for HTTP connections.
com.softwareag.mobile.runtime.nui.nUIListView com.softwareag.mobile.runtime.nui.nUIListElement com.softwareag.mobile.runtime.nui.IListRenderer com.softwareag.mobile.runtime.nui.IListProvider com.softwareag.mobile.runtime.nui.IListListener	New ListView API.
com.softwareag.mobile.runtime.RuntimeInfo com.softwareag.mobile.runtime.SystemInfo	Access to Runtime and System information.
com.softwareag.mobile.md.io.File	New File API for accessing local file systems

Added API	Description
<code>com.softwareag.mobile.md.net.ConnectivityStatusRequester</code> <code>com.softwareag.mobile.md.net.IConnectivityWatcher</code>	New Connectivity monitoring API.
<code>com.softwareag.mobile.md.imagetools.AssetImageLoader</code>	AssetImage API to work with new asset catalogs.
<code>com.softwareag.mobile.runtime.toolkit.operations.queue.BasicOperationQueue</code> <code>com.softwareag.mobile.runtime.toolkit.operations.queue.ConditionalOperationQueue</code>	New operation queueing API.
<code>com.softwareag.mobile.runtime.toolkit.transformer.ImageTransformer</code>	New image/asset transformer
<code>com.softwareag.mobile.runtime.toolkit.operations.cache.CacheFirstCacheHandler</code> <code>com.softwareag.mobile.runtime.toolkit.operations.cache.CacheOnlyCacheHandler</code> <code>com.softwareag.mobile.runtime.toolkit.operations.cache.FileCacheManager</code> <code>com.softwareag.mobile.runtime.toolkit.operations.cache.IOperationCacheHandler</code> <code>com.softwareag.mobile.runtime.toolkit.operations.cache.ServerFirstCacheHandler</code> <code>com.softwareag.mobile.runtime.toolkit.operations.cache.ServerOnlyCacheHandler</code>	New operation caching API
<code>com.softwareag.mobile.runtime.nui.nUITableElement</code> public void updateRelativeWidths(int[] new_rel_widths)	
<code>com.softwareag.mobile.runtime.nui.nUIWebView</code> <code>com.softwareag.mobile.runtime.nui.nUIWebviewElement</code> public String evaluateJavaScript(String script) public void evaluateJavaScript(String script, IJSCallback callback)	

Removed API	Replacement, if any
<code>com.softwareag.mobile.runtime.toolkit.ui.controller.AbstractViewController</code> boolean isVisible(final nUIDisplayObject element) boolean setVisible(final nUIDisplayObject element, final boolean isVisible) boolean forceHide(final nUIDisplayObject element)	<i>nUIDisplayObject</i> now has methods to change visibility.
<code>com.softwareag.mobile.runtime.toolkit.ui.controller.IResourceController</code> public Image loadImage(final String name, final int viewBackGroundcolor) public Image loadImage(final String name, final int viewBackGroundcolor, final int containerWidth)	

Deprecated API

Replacement, if any

com.softwareag.mobile.runtime.core.Application
com.softwareag.mobile.runtime.core.CanvasCore
com.softwareag.mobile.runtime.core.CanvasBase
com.softwareag.mobile.runtime.core.CanvasDimensions
com.softwareag.mobile.runtime.core.CanvasInterrupts
com.softwareag.mobile.runtime.core.CanvasMenu
com.softwareag.mobile.runtime.core.CanvasNativeUI
com.softwareag.mobile.runtime.core.CanvasSoftKeys
com.softwareag.mobile.runtime.core.CanvasThreading
com.softwareag.mobile.runtime.comms.HttpConnectionHandler

com.softwareag.mobile.runtime.toolkit.ui.controller.AbstractViewController

 protected String getGeneralGraphicsFolder(final int viewBackGroundcolor, final int containerWidth)
 protected Image tryLoadImage(final String image)
 protected String getPlatformGraphicsFolder(final int viewBackGroundcolor)
 protected String getWinPhoneGraphicFolder(final int width, final int height, final int viewBackGroundcolor)
 protected String getWinPhoneThemeFolder(final int backgroundColor) protected String getIOSGraphicsFolders(final int currentScreenPPI, final int viewBackGroundcolor)
 protected String getAndroidGraphicsFolder(final int currentScreenPPI, final int viewBackGroundcolor)

Changed API

Description

com.softwareag.mobile.runtime.toolkit.ui.controller.AbstractViewController

 protected Image loadImage(final String name, final int viewBackGroundcolor)
 protected Image loadImage(final String name, final int viewBackGroundcolor, final int containerWidth)

public to
protected

9.0 Copyright Information

Copyright © 2017 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

10.0 Support

Visit the [Empower website](#) to learn about support policies and critical alerts, read technical articles and papers, download products and fixes, submit feature/enhancement requests, and more.

Visit the [TECHcommunity website](#) to access additional articles, demos, and tutorials, technical information, samples, useful resources, online discussion forums, and more.

MOB-RM-101-20171017