

Release Process

Aptean Ltd
Copyright © 2011-2025.

Contents

- 1 Release Process.....1
 - 1.1 .NET.....1
 - 1.2 Android.....4
 - 1.3 Documentation.....5

1 Release Process

1.1 .NET

1.1.1 Build a Full Release

DevOps

- Complete all changes in branches and create all pull requests.
- Complete all pull requests into DEV
- Create a pull request from DEV to QA
- Complete the pull request from DEV to QA.

From Visual Studio

- From Team Explorer
 - ◆ Switch to QA Branch
 - ◆ From Sync tab, fetch and pull all changes into local QA branch.
- Update AssemblyInfo to next version
 - ◆ Open an assemblyInfo file.
 - ◆ Global Replace the version with the next version throughout the solution
 - ◆ Note: If also releasing Android, may also need to update UpdatesInfo.xml in WebApp project, \WebServices\Calidus_ePOD\PDUpdates folder.
- From Team Explorer
 - ◆ Stage and commit these changes with commit description "Version 4.x.y.z"
 - ◆ Sync and Push these changes.

From DevOps:

- Tag this commit in QA branch as tag "4.x.y.x" with appropriate description

From Visual Studio

- From Solution Explorer:
 - ◆ Rebuild Solution
 - ◆ Rebuild all others including import and export
 - ◆ Right-click WebApp and choose Publish
 - ◆ Set the Target location as a local folder (re: Publish folder).
 - ◆ Click Publish.
 - ◆ Note: There are sometimes issues with AjaxControlToolkit.pdb. If this is the case when publishing, copy the file from a prior release into your WebApp/bin folder and try to publish again.
 - ◆ Open Publish folder in File Explorer.
- From Publish Folder:
 - ◆ Delete web.config
 - ◆ Delete Images/Sites folder

From a local release folder:

- Copy a previous Release folder and rename with the version i.e. EPOD_4.x.y.z (re: Release folder)
- Empty web from Release folder.
- Copy all folders from Publish folder to Release/web folder
- Empty app from Release folder.
- Copy all folders and files from Publish folder to Release/app folder
- Create Auto_Import and Auto_Export folder in Release/app folder
- From Visual Studio, explore the Auto_Import project in File Explorer.
- From this file explorer, go to bin, go to Debug or Release (depending on how you published), and copy all contents to Release/app/Auto_Import folder.
- From Release/app/Auto_Import, delete app config file.
- From Visual Studio, explore the ePOD_AutoExport project in File Explorer.
- From this file explorer, go to bin, go to Debug or Release (depending on how you published), and copy all contents to Release/app/Auto_Export folder.
- From Release/app/Auto_Export, delete app config file.
- From Visual Studio, explore the Epod_Configuration project in File Explorer.
- From this file explorer, go to 4.x.0.0 folder and copy all contents to Release/Epod_Configuration folder.



- From Visual Studio, explore the EPOD_DatabaseScriptsLibrary project in File Explorer.
- From this file explorer, go to bin/Debug folder and copy all contents to Release/EPOD_DatabaseScriptsLibrary folder.
- From folder above release folder, create a zip of the release folder i.e. EPOD_4.x.y.z.zip
- Copy release zip to \\DGA1FS01OBS\Projects\Releases\EPOD\Server\Releases.
- Create release notes in \\DGA1FS01OBS\Projects\Releases\EPOD\Release Notes\EPD Admin\1. Mainline

1.1.2 Releasing a Full Release

Note: There may be multiple app and web boxes - this release needs to be completed to both boxes, although the database scripts need be run only once. This process will show releases to App and Web boxes. For a single release, follow the App box release procedure.

If the servers are accessible through a jump box:

- Log on to the jump box.
- Copy the release zip to the jump box.
- Use this jump box as the source from this point

App box release:

- Log on to the App server.
- Stop the Auto_Import and Auto_Export scheduled processes using the Task Scheduler if they are running on this machine.
- Copy the release zip to the app box, C:_INSTALL_FILES
- Use File Explorer to go to C:_INSTALL_EPOD
- Copy the last release script for that customer or latest version of release script to new release script for that customer and version. e.g. Deploy_Customer_4.5.1.19_BRC_TST.bat
- Edit the release script
 - ◆ Change the parameters in the script e.g.

```
SET "__BuildVer=EPOD_4.5.1.19"
SET "__CustomerCode=BRC"
SET "__Environment=TST"
SET "__CustomerName="BRC"
```

- ◆ Save the changes
- Right-click the script and run as administrator.
- Follow instructions in command window
- A new command window will open with the database scripts library script
- Press Return to install general scripts
- Press Return to install client specific scripts
- Note any errors with the scripts and resolve later.
- Exit database scripts library script command window
- Finish the Release script command window prompts and close that window.
- Check and update any app or web config files with new required settings for all apps released.
- Restart the Auto_Import and Auto_Export scheduled processes using the Task Scheduler if they are running on this machine.

From Database Server:

- Check the release scripts have worked successfully by checking the database Schema_Versions table.
- Resolve and script issues. Note that any fixes must be checked into the solution and the release updated.

Web Box release:

- Log on to the Web server.
- Stop the Auto_Import and Auto_Export scheduled processes using the Task Scheduler if they are running on this machine.
- Copy the release zip to the web box, C:_INSTALL_FILES
- Use File Explorer to go to C:_INSTALL_EPOD
- Copy the last release script for that customer or latest version of release script to new release script for that customer and version.
- Edit the release script



- ◆ Change the release version.
- ◆ Ensure customer and TST/PRD are set properly.
- ◆ Save the changes
- Right-click the script and run as administrator.
- Follow instructions in command window
- A new command window will open with the database scripts library script - close it.
- Finish the Release script command window prompts and close that window.
- Check and update any app or web config files with new required settings for all apps released.
- Restart the Auto_Import and Auto_Export scheduled processes using the Task Scheduler if they are running on this machine.


Update the release register \\DGA1FS01OBS\Projects\Releases\EPOD\Release Logs\Release Register - Master.xls

1.1.3 Correcting Script Issues

- Use DEV branch
- Find and fix the issues in script
- Stage and commit the changes
- Follow the steps above in section [Build a Full Release](#)
- Note that build versions need not be updated again, and tagged release should be appended with a annotation that this is a rebuild e.g. "4.05.01.19.R1" or "4.05.01.19.C1"

1.1.4 Rolling back a release

When following a full release procedure, a backup of the previous application and web installs is made. There is no automated process for re-installing this - simply copy the backup back over the app and web directories to roll back the release.

 **Warning:** This does not rollback database changes - these must be completed manually.

1.1.5 Releasing PDA

- Set the Debug/Release drop-down to Release.
- Build the ePODClientInstaller, ePODUpdaterInstaller and ePODClientInstallerComplete projects.
 - ◆ To build right click the project and select Build.
 - ◆ The ePODClientInstaller and ePODUpdaterInstaller projects must be built before the ePODClientInstallerComplete
- Copy the three cab files from your local ePOD_AutoExport\bin\Release folder (e.g. C:\Projects\EPOD\EPoD_WebApp\EPoD_AutoExport\bin\Release) to the PDA folder within the release directory (e.g. P:\Releases\EPOD\Server\EPoD PDA V1.31.1.9\PDA).
- To release:
 - ◆ Copy the three cab files from the release PDA directory to the PDAUpdates directory on the target machine. This directory is below the Admin web application directory in the WebServices\Calidus_ePOD subdirectory.
 - ◆ In the same PDAUpdates directory update the version numbers in the UpdatesInfo.xml file. Example:

```
<?xml version="1.0" encoding="utf-8" ?>
<UPDATES>
  <EPoD_CLIENT version="1.30.0.0"
    url="http://my.url.com:9093/WebServices/Calidus_ePOD/PDAUpdates/EPoDClientInstaller.CAB"
    required="Y" />
  <EPoD_CLIENT_ANDROID version="3.00.02.09"
    url="http://my.url.com:40462/WebServices/Calidus_ePOD/PDAUpdates/CALIDUSEPoDv3.apk"
    required="Y" />
  <EPoD_UPDATER version="1.26.2.0"
    url="http://my.url.com:9093/WebServices/Calidus_ePOD/PDAUpdates/ePODUpdaterInstall.CAB"
    required="Y" />
</UPDATES>
```



1.1.6 AutoExport/AutoImport

- Using visual studio get the latest version of the AutoExport or AutoImport processes.
- Use the publish option.
- Copy the contents of the build folder typically (Project\Release) to your release folder

- At this point this should be deployed to Dev and Test.

1.2 Android


1.2.1 PDA Client

Official releases for Android are through the Google Play market.

Google Play

Once the APK is built:

- Open Google developer site (https://play.google.com/apps/publish/v2/?dev_acc=06502315555851208789) and log in with the provided user and password.
- Choose the CALIDUS ePOD application by clicking on it.
- Click *APK* on the menu, and **Upload New APK**.
- Drag the new version of the APK into the provided box.
- When uploaded, click **Publish Now**.
- Click *Store Listing* on the menu.
- Add recent changes. This should be extracted from the latest entry in CHANGELOG.txt.
- Click **Save and Publish**

 **Note:** In the future it may be that each client will have its own version of the application on the market. If that is the case, ensure that the correct version is built and updated.

Hosted File

To host the file properly, this requires IIS to be updated as follows:

- Start IIS Manager on the server
- Find the hosted EPOD web site (or change top-level default)
- Edit MIME Types
- Add a new one, ".apk", type "application/vnd.android.package-archive".

Once the app is built, drop the app file in your_app_folder\webServices\calidus_epod\PDAUpdates\


Update the UpdatesInfo.xml file as follows:

```
<?xml version="1.0" encoding="utf-8" ?>
<UPDATES>
  <EPOD_CLIENT version="1.30.0.0"
    url="http://my.url.com:9093/WebServices/Calidus_ePOD/PDAUpdates/EPODClientInstaller.CAB"
    required="Y" />
  <EPOD_CLIENT_ANDROID version="3.00.02.09"
    url="http://my.url.com:40462/WebServices/Calidus_ePOD/PDAUpdates/CALIDUSePODv3.apk"
    required="Y" />
  <EPOD_UPDATER version="1.26.2.0"
    url="http://my.url.com:9093/WebServices/Calidus_ePOD/PDAUpdates/ePODUpdaterInstall.CAB"
    required="Y" />
</UPDATES>
```

Attributes:


- version - your build version, without any leading or trailing characters
- url - the url of where this installer may be accessed
- required - if this is a critical bug fix, set this to "Y", else "N"




 **Note:** Case is important!

Access to this APK is then through the url: http://your.url.com/webservices/calidus_epod/PDAUpdates/your.apk

1.3 Documentation

 **Note:** This section is not intended as a replacement for standard procedures, but instead a reminder of the responsibilities for releasers.

- Once programs have been packaged and released to the customer's test system, the affected Supimix Logs should be updated to reflect this - their status and sub status should be changed, the Patch number should be updated with the patch version and a comment added e.g. "Release to customer test system in patch 123456/version 1.2.3".
- Once programs have been packaged and released to the customer's live system, the affected Supimix Logs should be updated to reflect this - their status and sub status should be changed and a comment added e.g. "Release to customer live system in patch 123456/version 1.2.3". This comment should contain enough detail for a project resource to find the release notes in the release area or source control.

 **Note:** If multiple programs are being updated on the same log, this may result in multiple releases (i.e. a Device Client release and a Server release). As both can't be held in the patch number, the Server release should take precedence and the device client version number referenced in those release notes and in the Supimix comment.

