

Release procedures

The release procedure for an upcoming release is as following:

- All developments take place in the trunk until the code freeze. The code freeze will be at least 2 days before the release date
- On the day the code freeze becomes active, a release branch is created.
- The trunk and its assemblies will need to be updated, see "Updating the version number"
- A Teamcity configuration will be created for the release branch, see "Creating a TeamCity release configuration"
- Further developments related to the release are being further developed in the release branch. The changes are merged BACK to the trunk. This goes on until the release is approved
- On the day of release, a version is pinned on the release branch. This version will be checked briefly by either the P/O or the tester as a sanity check
- If the check is ok, the signed version is made and the release is shipped

Updating the manual

The following lines have to be updated also in the MorphAn.wixproj after the manual is placed and committed in the repository

```
<Content Include="Manuals\MorphAn Gebruikershandleiding 1.7.pdf" />  
<Content Include="Manuals\MorphAn User Manual 1.5.pdf" />  
first line = Dutch version  
second line = English version
```

Also the following files in setup/MorphAn/Localization/ need to be updated:

- en-US.wxl
- nl-NL.wxl

Replace the line

```
<String Id="ManualFileName" Overridable="yes">MorphAn User Manual 1.10.1.pdf</String>
```

with the corresponding manual file name

Updating the version number

Adjust the following files in the root at <https://repos.deltares.nl/repos/delft-tools/trunk/delta-shell/Products/MorphAn>

- AssemblyInfo.Version.cs
- AssemblyInfo.Version.WL.cs
- MorphAn.wixproj

Updating the Copyright year: (When applicable)

Besides the

- AssemblyInfo.Version.cs
- AssemblyInfo.Version.WL.cs

Also update the entry

```
<?define splashScreenCopyright="© Deltares 2022"?>
```

in the Product.wxs

Creating a TeamCity release configuration

1 Copy the most recent TeamCity subproject for MorphAn

Subprojects

Subprojects can be used to group build configurations and define projects hierarchy within a single project. [?](#)

[+ Create subproject](#)

sorted with custom order [Reorder](#)

Name	
Delta Shell - MorphAn Trunk	Edit ⋮
Delta Shell - MorphAn 1.9.X	Edit ⋮
Delta Shell - MorphAn 1.8.X	
Delta Shell - MorphAn 1.7.X	
Delta Shell - MorphAn 1.6.X	
Delta Shell - MorphAn 1.5.X	
Delta Shell - MorphAn 1.4.X	
Delta Shell - MorphAn 1.3.X	
Delta Shell - MorphAn 1.2.3	
Delta Shell - MorphAn 1.2.X	
Delta Shell - MorphAn 1.1.X	Edit ⋮
Delta Shell - MorphAn 1.0.X	Edit ⋮
Delta Shell - MorphAn 0.3 (archived)	Edit ⋮
13-08-2012 release	

[Copy project...](#)
[Move project...](#)
[Archive project...](#)
[Bulk edit IDs...](#)
[Pause/Activate...](#)
[Scramble secure value...](#)
[Export project...](#)
[Download settings in Kotlin format...](#)
[Delete project...](#)

TeamCity Enterprise 2020.1.2 (build 78726)

[New version is available!](#)

[License agreement](#)

2 Adjust the checkout path in the parameters of the copied configuration. Point this path to the newly created release branch

Delta Shell - MorphAn 1.9.X

[⚠ 4 build agents running git version prior to 2.10.0.0, which will be no longer supported starting from the next major TeamCity release. Update git executable on agents.](#) [Hide](#)
[Show details](#)

[General Settings](#)
[VCS Roots](#)
[Report Tabs](#)
[Parameters](#)
[Builds Schedule](#)
[Connections](#)
[SonarQube Servers](#)
[Shared Resources](#)
[SSH Keys](#)
[Meta-Runners](#)
[Maven Settings](#)
[Issue Trackers](#)
[Cloud Profiles](#)
[Clean-up Rules](#)
[Versioned Settings](#)
[Artifacts Storage](#)
[NuGet Feed](#)
[Suggestions](#)
[Hide unconfigured](#)

[+ Add new parameter](#)
☐ Show inherited parameters
There is **1** own parameter defined and **7** inherited
Configuration Parameters
Configuration parameters are not passed into build, can be used in references only. [?](#)

Name	Value	
checkout_path	branches/delta-shell-morphan-1.9	Edit Delete

System Properties (system.)
System properties will be passed into the build (without system. prefix), they are only supported by the build runners that understand the property notion. [?](#)
None defined
Environment Variables (env.)
Environment variables will be added to the environment of the processes launched by the build runner (without env. prefix). [?](#)

Last edited 3 days ago
by Dennis Tang [\(view history\)](#)