

```
1 function revSBEnginePath pPlatform
2   local tEdition
3   put the editionType into tEdition
4   // SN-2015-07-09: [[ StandaloneDeployment ]] Gypified Standalone-<edition>.app building does not
5   // allow to create broken applications, so that the executable file in it bears the edition.
6   if tEdition is not "community" then
7     put "Commercial" into tEdition
8   end if
9
10  -- MW-2013-06-13: [[ CloneAndRun ]] If not installed, then fetch the engine locally.
11  if revEnvironmentIsInstalled() then
12    # MW-2009-06-24: For Mac OS X, engine source is two lines, first is PPC (if any), second
13    # is x86 (if any). Both implies universal build.
14    -- MM-2014-03-21: [[ PPC Support Dropped ]] We now only support intel Mac builds. Assume all mac builds to be intel.
15    local tRuntimePath
16    put revEnvironmentRuntimePath() into tRuntimePath
17    switch pPlatform
18      case "MacOSX"
19      case "MacOSX PowerPC-32"
20      case "MacOSX x86-32"
21
22      case "MacOSX x86-64"
23        // SN-2015-07-09: [[ StandaloneDeployment ]] Gypified Standalone-<edition>.app building does not
24        // allow to create broken applications, so that the executable file in it bears the edition.
25        put the upper of char 1 of tEdition into char 1 of tEdition
26
27        return tRuntimePath & "/Mac OS X/x86-32/Standalone.app/Contents/MacOS/Standalone-" & tEdition
28
29        break
30      case "Windows"
31        return tRuntimePath & "/Windows/x86-32/Standalone"
32        break
33      case "Windows x86-64"
34        return tRuntimePath & "/Windows/x86-64/Standalone"
35        break
36      case "Linux"
37        return tRuntimePath & "/Linux/x86-32/Standalone"
38        break
39      -- MW-2013-11-06: [[ LinuxX64 ]] Compute the correct path for 64-bit linux.
40      case "Linux x64"
41        return tRuntimePath & "/Linux/x86-64/Standalone"
42        break
43      -- FG-2014-08-19: [[ RPi ]] Compute the correct path for ARMv6 Linux
44      case "Linux armv6-hf"
45        return tRuntimePath & "/Linux/armv6-hf/Standalone"
46        break
47      default
48        return "error"
49        break
50    end switch
51  else
52    -- Make sure the platform is correct
53    // SN-2015-05-15: [[ DeployAllPlatforms ]] Allow the deployment to happen
54    // for another platform, from the repository.
55    local tBinariesPath
56
57    if pPlatform contains "MacOSX" and the platform is not "macos" or \
58      pPlatform contains "Linux" and the platform is not "linux" or \
59      pPlatform is "Windows" and the platform is not "win32" then
60      put revEnvironmentNonNativeBinariesPath(pPlatform) into tBinariesPath
61    else
62      put revEnvironmentBinariesPath() into tBinariesPath
63    end if
64
65    switch word 1 of pPlatform
66      case "MacOSX"
67        put the upper of char 1 of tEdition into char 1 of tEdition
68        return tBinariesPath & "/Standalone-" & tEdition & ".app/Contents/MacOS/Standalone-" & tEdition
69        break;
70      case "Windows"
71        return tBinariesPath & slash & "standalone-" & tEdition & ".exe"
72        break
73      case "Linux"
74        return tBinariesPath & slash & "standalone-" & toLower(tEdition)
75        break
76      default
77        return empty
78        break
79    end switch
80  end if
81 end revSBEnginePath
```

```
1 function revSBEnginePath pPlatform
2   local tEdition
3   put the editionType into tEdition
4   // SN-2015-07-09: [[ StandaloneDeployment ]] Gypified Standalone-<edition>.app building does not
5   // allow to create broken applications, so that the executable file in it bears the edition.
6   if tEdition is not "community" then
7     put "Commercial" into tEdition
8   end if
9
10  -- MW-2013-06-13: [[ CloneAndRun ]] If not installed, then fetch the engine locally.
11  if revEnvironmentIsInstalled() then
12    # MW-2009-06-24: For Mac OS X, engine source is two lines, first is PPC (if any), second
13    # is x86 (if any). Both implies universal build.
14    -- MM-2014-03-21: [[ PPC Support Dropped ]] We now only support intel Mac builds. Assume all mac builds to be intel.
15    local tRuntimePath
16    put revEnvironmentRuntimePath() into tRuntimePath
17    switch pPlatform
18      case "MacOSX"
19      case "MacOSX PowerPC-32"
20      case "MacOSX x86-32"
21
22      case "MacOSX x86-64"
23        // SN-2015-07-09: [[ StandaloneDeployment ]] Gypified Standalone-<edition>.app building does not
24        // allow to create broken applications, so that the executable file in it bears the edition.
25        put the upper of char 1 of tEdition into char 1 of tEdition
26        return tRuntimePath & "/Mac OS X/x86-32/Standalone.app/Contents/MacOS/Standalone-Community" -- & tEdition
27        break
28
29      case "MacOSX x86-64"
30        // SN-2015-07-09: [[ StandaloneDeployment ]] Gypified Standalone-<edition>.app building does not
31        // allow to create broken applications, so that the executable file in it bears the edition.
32        put the upper of char 1 of tEdition into char 1 of tEdition
33
34        return tRuntimePath & "/Mac OS X/x86-64/Standalone.app/Contents/MacOS/Standalone-Community" -- & tEdition
35
36        break
37      case "Windows"
38        return tRuntimePath & "/Windows/x86-32/Standalone"
39        break
40      case "Windows x86-64"
41        return tRuntimePath & "/Windows/x86-64/Standalone"
42        break
43      case "Linux"
44        return tRuntimePath & "/Linux/x86-32/Standalone"
45        break
46      -- MW-2013-11-06: [[ LinuxX64 ]] Compute the correct path for 64-bit linux.
47      case "Linux x64"
48        return tRuntimePath & "/Linux/x86-64/Standalone"
49        break
50      -- FG-2014-08-19: [[ RPi ]] Compute the correct path for ARMv6 Linux
51      case "Linux armv6-hf"
52        return tRuntimePath & "/Linux/armv6-hf/Standalone"
53        break
54      default
55        return "error"
56        break
57    end switch
58  else
59    -- Make sure the platform is correct
60    // SN-2015-05-15: [[ DeployAllPlatforms ]] Allow the deployment to happen
61    // for another platform, from the repository.
62    local tBinariesPath
63
64    if pPlatform contains "MacOSX" and the platform is not "macos" or \
65      pPlatform contains "Linux" and the platform is not "linux" or \
66      pPlatform is "Windows" and the platform is not "win32" then
67      put revEnvironmentNonNativeBinariesPath(pPlatform) into tBinariesPath
68    else
69      put revEnvironmentBinariesPath() into tBinariesPath
70    end if
71
72    switch word 1 of pPlatform
73      case "MacOSX"
74        put the upper of char 1 of tEdition into char 1 of tEdition
75        return tBinariesPath & "/Standalone-" & tEdition & ".app/Contents/MacOS/Standalone-" & tEdition
76        break;
77      case "Windows"
78        return tBinariesPath & slash & "standalone-" & tEdition & ".exe"
79        break
80      case "Linux"
81        return tBinariesPath & slash & "standalone-" & toLower(tEdition)
82        break
83      default
84        return empty
85        break
86    end switch
87  end if
88 end revSBEnginePath
```