

## Release Note CODESYS V3.5 SP13 Patch 1

23.08.2018

### 1 Release Notes

Key	Summary	Release Note	Component/s
<a href="#">CDS-55475</a>	Setup Redesign	<p><b>[[COMPATIBILITY_INFORMATION-OEM]]</b>            For more information on creating OEM installations, please refer to the new documentation "CODESYS Installation Extended OEM Adaptions"</p> <p><b>[[COMPATIBILITY_INFORMATION-EndUser]]</b>            The default installation folder has changed to            &lt;ProgramFiles(x86)&gt;\CODESYS            &lt;version&gt; or            &lt;ProgramFiles&gt;\CODESYS            &lt;version&gt;.            Each installation has to be installed in a new folder. It is no longer allowed to install to an already existing installation.            For more information on calling and adapting the installations, please refer to the new documentation "CODESYS Installation OEM Adaptions"</p>	CODESYS
<a href="#">CDS-11695</a>	<b>Symbolconfiguration:</b> Configurable set of different symbols for separate clients must be supported	<p><b>[[COMPATIBILITY_INFORMATION]]</b>            To use this new feature, you need:            - CODESYS &gt;= v3.5.13.0:              o CODESYS CompilerVersion v3.5.13.0 (and so lecVarAccess.library and SymbolicVarsBase.library v3.5.13.0 are used)              o New Symbolconfig Editor              o Symbol Set Editor            - Runtime System &gt;= v3.5.13.0</p> <p><b>[[COMPATIBILITY_INFORMATION-EndUser]]</b>            Because of changes made to the library structure, you must not mix up</p>	CODESYS, CODESYS Control

		<p>library placeholders manually in the following versions:</p> <ol style="list-style-type: none"> <li>1. lecVarAccess, SymbolicVarsBase, DataSources * Versions &lt; V3.5.13.0 and Version &gt;= V3.5.13.0</li> <li>2. CompilerVersion &lt; V3.5.13.0 and lecVarAccess, SymbolicVarsBase, DataSources * &gt;= V3.5.13.0</li> </ol> <p>For the implementation of this new feature it was necessary to completely revise the algorithm to resolve the symbolic variable names. The new algorithm analyzes the variable name more strictly, so that, for example, spaces at the end of the symbolic variable name are no longer allowed. As a result the new algorithm does not only support the new symbol set feature, but also reduce the time to resolve variable names significantly.</p> <p>It may happen that some old project contains the TraceManager library. This library is not used by current visualizations. This causes compile errors. To fix this:</p> <ul style="list-style-type: none"> <li>- Remove the TraceManger library by hand</li> </ul> <p>If there are compile errors caused by the trace editor:</p> <ul style="list-style-type: none"> <li>- Open the trace editor. Change something (e.g. activate and deactivate a trigger) and close the editor.</li> </ul>	
<p><a href="#">CDS-53589</a></p>	<p><b>Auto Declare:</b> Autodeclaration does not function as expected, because "instance paths" are required by default</p>	<p><b>[[KNOWN_LIMITATIONS]]</b> Auto Declare cannot add namespaces after declaring variables in qualified only GVLs. This is a known limitation and it will be resolved in the future with CDS-52466. For now Auto Declare will not declare an already existing variable in such a GVL a second time and avoid further error message with that.</p>	<p>CODESYS</p>

<a href="#">CDS-57134</a>	<p><b>Compile Compatibility:</b> Download required for specific project</p>	<p><b>[[COMPATIBILITY_INFORMATION]]</b> This issue can only be fixed by changing an incorrect existing compiler version check. This will lead to new incompatibilities. CODESYS V3.5 SP13 will be compatible to projects created with V3.5 SP4. But V3.5 SP13 will not be incompatible to projects created between V3.5 SP5 - V3.5 SP12 when the compiler version V3.5 SP4 is used in those projects. The decision was made that the compatibility to actual V3.5 SP4 projects is more important than the second use case.</p>	<p>CODESYS</p>
<a href="#">CDS-57693</a>	<p><b>LMM:</b> Static Area: Not possible to relocate in CODESYS</p>	<p><b>[[COMPATIBILITY_INFORMATION-OEM]]</b> Due to an error in the compiler, the setting StartAddress in the target settings was not used for the static area. With Compiler version <math>\geq 3.5.13.0</math> the setting will be used for code generation. An accidental setting of this value in previous versions could lead to a crash at download.</p>	<p>CODESYS</p>

<p><a href="#">CDS-58134</a></p>	<p>Add compiler version 3.5.3.87 to be available in newer versions</p>	<p><b>[[GENERAL]]</b>  The new Compiler Version has no effect on the generated code, the generated code will be the same as for Compiler Version 3.5.3.86.  The version of CODESYS 3.5.3.87 and the current version have diverged too much, <b>so that it is not possible to guarantee that the same code is generated.</b></p> <p>However: a project created with compiler 3.5.3.87 can be opened with a new CODESYS and will not require a rebuild. As long as nothing is changed in the project, and if the old compile-info still exists, a login to a running system should be possible. We do not recommend the usage of this compiler version for new builds. If you change your project, you should consider an update of the compiler version</p>	<p>CODESYS</p>
----------------------------------	--	--	----------------

<p><a href="#">CDS-58447</a></p>	<p><b>Compiler:</b> Unexpected online change in specific project with new CODESYS version</p>	<p><b>[[COMPATIBILITY_INFORMATION]]</b> The problem only occurs for persistent variable lists that are decorated with an attribute. The problem was introduced with Codesys Version 3.5.10.0. With this version the internal order of a list of attributes of an object might change, without respect to the compiler version. Therefore projects with CODESYS Version &gt; 3.5.10.0 may produce a language model for Persistent vars (with additional attribute) that is different to previous versions. We now generate for projects with compilerversion &lt; 3.5.10.0 a compatible the same list as with codesys &lt; 3.5.10.0. Projects with compilerversion &lt; 3.5.10.0, that were created with codesys versions &gt;= 3.5.10.0 may now be different! This is a problem that we can't avoid since we only know the compiler version and not the codesys version of the created project.  For CompilerVersion &gt;= 3.5.13.0 the order of the attributes is always fix (sorted lexically).</p>	<p>CODESYS</p>
----------------------------------	---	---	----------------

<p><a href="#">CDS-56864</a></p>	<p><b>Automation Platform:</b> Create a local GAC and do not use Windows GAC anymore</p>	<p><b>[[GENERAL]]</b> CODESYS is now able to load strong named assemblies (GAC components) from a local directory within the installation folder. The layout will be as follows: &lt;RootDirectory&gt;/&lt;LacBinaries&gt;/&lt;GAC_MSIL&gt;/&lt;AssemblyName&gt;/&lt;Version&gt;__&lt;PublicKeyToken&gt;/&lt;AssemblyName&gt;.dll</p> <p>Beside GAC_MSIL there can be folders for 32 bit (GAC_32) and 64 bit (GAC_64) architectures. This is equivalent to the global assembly cache. The local assembly cache will only be queried when the assembly could not be found within the global assembly cache.</p> <p>Executables that are using CODESYS GAC components have to redirect the assembly resolution to the new local directory 'LacBinaries'. The redirection can be achieved using an assembly binding within the exe's configuration file. The following example redirects to Control.dll to the new directory:</p> <pre>&lt;dependentAssembly&gt;   &lt;assemblyIdentity     name="Controls"       publicKeyToken="83380e73b2486719"       culture="neutral" /&gt;   &lt;codeBase version="3.1.1.2"     href="..\LacBinaries\GAC_MSIL\Controls\3.1.1.2__83380e73b2486719\Controls.dll"/&gt; &lt;/dependentAssembly&gt;</pre> <p><b>[[KNOWN_LIMITATIONS]]</b> If all assemblies will be stored within the global assembly cache, the local assembly cache will never become active.</p>	<p>CODESYS</p>
----------------------------------	--	--	----------------

<a href="#">CDS-59648</a>	<p><b>Visu:</b> No login without change when using compiler versions &lt; 3.5.9</p>	<p><b>[[COMPATIBILITY_INFORMATION]]</b> To fix this issue it was necessary to introduce a minor incompatibility under special circumstances when using target- or webvisualziation within a project. It will no longer be possible to login to applications without change under the following circumstances: * a project containing target- or webvisualization was downloaded * a compilerversion &gt;= 3.5.7 and &lt; 3.5.9 was used during this download * the download was done using a programming system &gt;= 3.5.12 and &lt; 3.5.12.30</p> <p>This is a problem that we can't avoid since we only know the compiler version and not the codesys version of a downloaded project.</p>	<p>CODESYS</p>
<a href="#">CDS-56007</a>	<p><b>Profinet Controller (CIFX + CIFX V3):</b> Firmware files and devdescs</p>	<p><b>[[COMPATIBILITY_INFORMATION]]</b> Default version of Profinet Controller firmware (file 'cifxpmn.nxf') changed to version V3.x. Either update the profinet controller device in your project (recommended) or use obsolete firmware (file 'cifxpmn2.nxf').</p>	<p>CODESYS</p>
<a href="#">CDS-45548</a>	<p><b>FEATURE Multicore support:</b> The CODESYS Runtime must support Multicore architectures</p>	<p><b>[[COMPATIBILITY_INFORMATION]]</b> 1. All IEC task are running on one CPU core by default! So here we have no difference to a runtime system before v3.5.13.0 2. All other lower priority tasks (Monitoring / Server Task) are running free floating on the operating system! So these task can run on different CPU cores as the IEC tasks! As a consequence, writing an IEC value or reading an IEC value could occur right in the middle of executing the IEC tasks! This was still the case on a single core</p>	<p>CODESYS Control</p>

		<p>CPU, but now we have a concurrent execution of monitoring an IEC task execution!</p> <p>If this is a problem in your environment, you can bind the complete runtime system on one core with a cfg-Setting of the runtime system, e.g. to bind on core 1 (index 0 based):  [SysCpuMultiCore]  BindProcess.CoreID=1</p>	
<a href="#">CDS-53155</a>	<b>CmpOpenSSL:</b> Update to OpenSSL 1.0.2o	<p><b>[[GENERAL]]</b>  For more details see Advisory 2018-06, which is available on the CODESYS website:  <a href="https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-06_CDS-53155.pdf">https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-06_CDS-53155.pdf</a></p>	CODESYS Control
<a href="#">CDS-58208</a>	<b>CmpOPCUAServer:</b> Crash when invalid request is sent	<p><b>[[GENERAL]]</b>  For more details see Advisory 2018-02, which is available on the CODESYS website:  <a href="https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-02_CDS-58208.pdf">https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-02_CDS-58208.pdf</a></p>	CODESYS Control
<a href="#">CDS-56118</a>	<b>CodeMeter SDK - CmpCodeMeter:</b> Separation from the runtime kernel and move to Contrib repository	<p><b>[[COMPATIBILITY_INFORMATION-OEM]]</b>  CodeMeterEmbedded source files are separated now completely from the runtime source files!  So maybe the OEM specific build process must be adapted to this new source file structure, if the OEM build the runtime from source!</p>	CODESYS Control
<a href="#">CDS-58127</a>	<b>SysCpuMultiCore:</b> New plcshell command necessary to display all tasks from SysTask	<p><b>[[COMPATIBILITY_INFORMATION-EndUser]]</b>  There is a new PlcShell command "gettasks" to display all tasks which are created via SysTaskCreate() resp. SysTaskCreate2() in the runtime system.</p>	CODESYS Control



<p><a href="#">CDS-59017</a></p>	<p><b>CmpFileTransfer:</b> Possibility of unauthorized file access to all system files</p>	<p><b>[[GENERAL]]</b> For more details see Advisory 2018-04, which is available on the CODESYS website: <a href="https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-04_CDS-59017.pdf">https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-04_CDS-59017.pdf</a></p> <p><b>[[COMPATIBILITY_INFORMATION]]</b> A new security feature is implemented to secure online file and directory access. It is represented by two new settings in [SysFile] section:</p> <p>ForceOnlineFilePath=1 (default) Setting to force the configured file path to every online file access. If an absolute file path is requested, which is not a configured path, or a directory traversal path an error is returned at this operation (ERR_OPERATION_DENIED). NOTE: ForceFilePath=1 dominates this setting.</p> <p>DenyOnlineAccessCfgFile=1 (default) Setting to deny online access to all configuration files. If a configuration file is requested, an error is returned at this operation (ERR_OPERATION_DENIED). NOTE: This setting is independent of any Force settings.</p> <p>To restore the old behaviour these settings may be configured as follows: [SysFile] ForceOnlineFilePath=0 DenyOnlineAccessCfgFile=0</p> <p><b>BUT WE HIGHLY RECOMMEND, TO LEAVE THESE SETTINGS AT THEIR DEFAULT VALUES!</b> For more information about configuration of filepath and placeholders see our corresponding tutorial as part of the CODESYS Control Runtime System Toolkit.</p>	<p>CODESYS Control</p>
----------------------------------	--	---	------------------------

<a href="#">CDS-55064</a>	<p><b>Linux / SysEthernet:</b> Make the use of QDISC_BYPASS configurable</p>	<p><b>[[COMPATIBILITY_INFORMATION-OEM]]</b> Linux Targets which support the PACKET_QDISC_BYPASS socketoption (usually kernels since 3.14) now have to set [SysEthernet] Linux.PACKET_QDISC_BYPASS=1 explicitly in the Runtime Configuration file to achieve the same behaviour of SysEthernet as in SP10 -12</p>	<p>CODESYS Control</p>
<a href="#">CDS-58820</a>	<p><b>CmpWebServer:</b> Possible run out of file descriptors</p>	<p><b>[[GENERAL]]</b> For more details see Advisory 2018-05, which is available on the CODESYS website: <a href="https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-05_CDS-58820.pdf">https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-05_CDS-58820.pdf</a></p>	<p>CODESYS Control</p>
<a href="#">CDS-52542</a>	<p><b>WinCE:</b> option to suppress runtime window</p>	<p><b>[[COMPATIBILITY_INFORMATION-OEM]]</b> There is an existing setting with a similar behavior: [ComponentManager] WindowHided=1 With setting enabled, the runtime (or HMI, or remote visu client) does not have the normal big runtime window, instead there is a tray icon with a very small menu (Show Info, and Exit/Close). Every OEM has source code for the runtime windows and can replace, remove or modify it.</p>	<p>CODESYS Control</p>

<a href="#">CDS-59124</a>	<b>WinCE:</b> SysTimeRtcSet should flush registry to make the new DT persistent	<b>[[COMPATIBILITY_INFORMATION-EndUser]]</b> Under WinCE, functions SysTimeRtcSet and SysTimeRtcHighResSet now perform a FlushRegistry (when time could be set successfully)	CODESYS Control
<a href="#">CDS-58440</a>	<b>WinCE:</b> separate SysReadWriteLock and SysCpuMulticore	<b>[[COMPATIBILITY_INFORMATION-OEM]]</b> Windows CE 7 and 8 runtimes now have dynamic SysReadWriteLock and SysCpuMultiCore components. Implementation of ReadWriteLocks is native.	CODESYS Control
<a href="#">CDS-52163</a>	<b>Webserver, SSL:</b> Connections can be blocked by a client	<b>[[GENERAL]]</b> For more details see Advisory 2018-05, which is available on the CODESYS website: <a href="https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-05_CDS-58820.pdf">https://customers.codesys.com/fileadmin/data/customers/security/2018/Advisory2018-05_CDS-58820.pdf</a>	CODESYS Control
<a href="#">CDS-56372</a>	<b>WinCE:</b> Generate lists of source files and components dynamically from selected features	<b>[[COMPATIBILITY_INFORMATION-OEM]]</b> WinCE runtime toolkits: Some of the .c files which used to be in the VC project of the CODESYS runtime system have now been replaced by .obj files. The .c files are still part of the delivery and can be used instead of the .obj files if needed.	CODESYS Control
<a href="#">CDS-58223</a>	<b>BACnet:</b> CmpBACnet - log BACstack messages in CODESYS	<b>[[GENERAL]]</b> IEC function CmpBACnet.BACnetEnableStackLogging can be used to enable/disable the log information of the BACnet-stack.	CODESYS Control

<a href="#">CDS-57585</a>	<b>BACnet:</b> CmpBACnet setting for BACNET_SRVR_INIT->appName	<b>[[GENERAL]]</b> codesyscontrol for linux CmpBACnet needs to create resources in the file system, default directory is /dev . Running codesyscontrol for linux without root permissions or without "create permissions" at /dev requires to set the [CmpBACnet]-setting "AppName=/tmp/CODESYSBACnetServer" or similar to point CmpBACnet to an appropriate directory.	CODESYS Control
<a href="#">CDS-56981</a>	<b>BACnet:</b> Integrate/Use new BACnet-Stack version (Rev 14)	<b>[[GENERAL]]</b> CmpBACnet 3.5.13.* supports BACnet protocol revision 14 now.  <b>[[COMPATIBILITY_INFORMATION]]</b> CmpBACnet 3.5.13.* requires IEC-lib-BACnet >= 1.2.1.0. IEC-lib-BACnet < 1.2.1.0 will not work with CmpBACnet 3.5.13.*.	CODESYS Control
<a href="#">CDS-60997</a>	<b>RTE Setup:</b> Prevent major upgrade to 3.5.13.0 and replace dynamic by static file references	<b>[[COMPATIBILITY_INFORMATION]]</b> A Major Upgrade of the RTE installation to previous versions is not possible. An older version has to be uninstalled before installing RTE 3.5.13.0.	CODESYS Control RTE
<a href="#">CDS-55940</a>	<b>Visu, UserManagement:</b> Wrong extension of interface IExternalUserDatabaseProvider	<b>[[COMPATIBILITY_INFORMATION]]</b> To fix this problem it was necessary to introduce a new interface library "VisuUserMgmt3 Interfaces". Projects or libraries that directly reference the former libraries "VisuUserMgmt2 Interfaces" or "VisuUserMgmt Interfaces" have to upgrade these references to "VisuUserMgmt3 Interfaces" when upgrading the compiler version to 3.5.13.0.  <b>[[GENERAL]]</b> This issue was fixed with VisuUserMgmt library >= 3.5.13.0 and therefore compiler version >= 3.5.13.0 is necessary.	Libraries



## 2 OEM information from JIRA

To read up on implemented features and changes you can use your JIRA account. Please find some **example** filters below.

### List of features and changes:

```
fixVersion = "V3.5 SP13 Patch 1"
```

```
fixVersion in ("V3.5 SP13", "V3.5 SP13 Patch 1") AND issuetype = "New Feature"
```

### List of features and changes since CODESYS V3.5 SP13:

```
fixVersion IN ("V3.5 SP13", "V3.5 SP13 Patch 1")
```

### List of issues with compatibility information and known limitations:

```
fixVersion in ("V3.5 SP13", "V3.5 SP13 Patch 1") AND (text ~ COMPATIBILITY_INFORMATION OR text ~ KNOWN_LIMITATIONS)
```

## 3 History

Created: Rico Ottliczky (Quality Assurance)  
Reviewed: Benjamin Schurr (Conifguration Management)  
Released: Benjamin Schurr (Conifguration Management)