Release Note CODESYS V3.5 SP10 Patch 10

08.03.2018

1 Compatibility information and known limitations

1.1 CODESYS

CDS-58561 Compile, IO Config: Wrong Task deployment for function blocks used in initial values

COMPATIBILITY_INFORMATION

With compiler version >= 3.5.10.100 only those tasks will be used as update task for instances with IOs in which an Access to the IOs can be detected, or in which the instance is called directly.

The behaviour is now again the same as with versions < V3.5.10.30. Beginning with this version for the fix of CDS-53189, we also used the declaration position for updating IOs. That could mean, that an Output was in unexpected tasks.

For example, if two instances with mapped IOs are declared in one GVL, and one of the instances is called in a task X, the output of the other instance would also be updated in task X.

The new behaviour, and the pre V3.5.10.30 behaviour could lead to less updates of Outputs as in the versions in between. For example, if an instance of a function block is declared in Program, but there is no direct call of the instance, there will be no update in the task in which the Program is called.

Direct call means in this case, that a call via an interface is not enough. A workaround for this problem is to manually edit the tasks in which to update IOs in the IO-Configuration.

CDS-58892 Compiler: Unexpected online change in specific project with new CODESYS version

COMPATIBILITY_INFORMATION

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 >= 3.5.10.100 AND < 3.5.11.0 may produce a language model for Persistent vars (with additional attribute) that is different to previous versions.

We now generate for projects with compilerversion < 3.5.10.0 the same list as with codesys < 3.5.10.0. Projects with compilerversion < 3.5.10.0, that were created with codesys versions >= 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 >= 3.5.10.100 and < 3.5.11.0 the order of the attributes is always fix (sorted lexically).

CDS-53365 LMM: Check all pool objects and Compile should report properties of type reference

COMPATIBILITY_INFORMATION-OEM

From Compilerversion 3.5.10.20 onwards the compiler also outputs the warning C0410 for the declaration of the property. This allows to also check libraries with ""Check all pool objects"" for this compatibility issue.

Once a property is adapted to the new behaviour the warning can be disabled by addign the following pragma to the declaration:

{attribute 'suppress_wrn_C0410'}"

CDS-52898   All setups require at least Windows 7 Service Pack 1

COMPATIBILITY_INFORMATION

System requirements are Windows 7 SP1 or newer

CDS-47515   CODESYS HMI SL

COMPATIBILITY_INFORMATION-EndUser

With CODESYS Version V3.5 SP10 the DataServer will be replaced by the new Data Sources Manager. Projects with DataServer can be used furthermore when the compiler version and the visualization profile is not updated. When the compiler version and the visualization profile is updated to V3.5 SP10 then the DataServer must be converted to the new Data Sources Manager. A command is provided in the context menu when the DataServer is selected. Please check the Known restrictions below too.

KNOWN_LIMITATIONS

Known restrictions in the new Data Sources Manager:

- CODESYS Application V3 Driver: Cross byteorder monitoring of STRING/WSTRING not possible (CDS-52563)
- Increased amount of generated code in some projects (CDS-52565)
- Data Sources Manager cannot be used with Device Application on a 64 bit target (CDS-52573)

If you experience such problems, please contact support@codesys.com

CDS-51296   ApplicationComposer

COMPATIBILITY_INFORMATION-EndUser

From Compilerversion 3.5.10.0 onwards an online change is no longer possible if the type of a variable is changed. This only applies to structured types like DUT and FB.

Furthermore an online change is also not possible anymore if the basetype of an FB or DUT is changed (the expression after the EXTENDS keyword.)

CDS-49939   AC: PersistenceManager: Use $ac_persistence$ placeholder

COMPATIBILITY_INFORMATION

For runtime version greater or equal 3.5.8.0 the place of archives used by the PersistenceManager has changed to placeholder $ac_persistence$. Existing archives are moved there when found in the former place $PlcLogic$. If an archive is found at the new path as well as at the old path then there is a warning in the log.

CDS-49943   SymbolConfig: Properties with monitoring type 'variable'

COMPATIBILITY_INFORMATION-EndUser

Starting with Compiler Version V3.5.10.0, properties with monitoring type ""variable"" are exported read-only in the Symbol Configuration. Writing them makes no sense, as the value is written to the shadow variable which is ignored by the IEC code, and it's even dangerous, as the monitoring code (debugger) also accesses the shadow variable until the property is called by the IEC code again, pretending wrong facts.

CDS-53134 ScriptEngine: Upgrade to IronPython 2.7.7 released version

GENERAL

IronPython was updated to the final 2.7.7 release, solving the memory leak problems and other bugfixes. There were no problems related to the Scripting except the compatibility issue with the Byte order marks (BOM).

COMPATIBILITY_INFORMATION

Due to bugfixes in upstream IronPython (compatibility to cPython), Byte order marks (BOM) at the start of files are not implicitly skipped any more. This might break some existing scripts which parse files, e. G. using the ConfigParser module. The recommended workaround is to open the file using codecs.open().

CDS-22416 CAN: It should be possible to use more than one CAN stack on the same CANBus

COMPATIBILITY_INFORMATION

Code Size of CANopen Stack 3.5.10.0 increased (e.g. 7 KB for X86, 15 KB for C167) due to some internal changes.

CDS-50823 LMM, Compile: POINTER TO PRG declaration should report an error

GENERAL

Change is effective for CompilerVersion >= 3.5.10.0

COMPATIBILITY_INFORMATION

A POU of Type Program does not have an associated connected data block, therefore a declaration of a POINTER TO PRG will now produce an error message.

Note that ADR(PRG) is possible, but does not return a POINTER TO the data of the Program, but a POINTER TO the Function pointer to the code of PRG.

If you think you need a Pointer to the program code, then you should declare a pointer like this:

```
pPrg: POINTER TO POINTER TO Byte;
```

And assign:

```
pPrg := ADR(PRG);
```

CDS-43667 CODESYS: LanguageModelManager LMM: structureinitializationexpressions have wrong parse tree

COMPATIBILITY_INFORMATION-EndUser

For Compiler Version >= 3.5.10.0 the initialisation order of arrays of User defined types changes:

e.g: for an array

```
arr : ARRAY [0..1] OF DUT := [(a := 1), (a := 2)];
```

Old order:

```
arr[0].FB_Init();
arr[0].a := 1;
arr[1].FB_Init();
arr[1].a := 2;
```

New order:

```
FOR i := 0 TO 1 DO
arr[i].FB_Init;
END_FOR
arr[0].a := 1;
arr[1].a := 2;
```

Furthermore if a user defined type is initialized by value-assigning an other variable like this:

```
xx : DUT := yy;
```

No call to FB_Init was produced. With Compiler Version >= 3.5.10.0 there will now also be a FB_Init call in this case.

CDS-44346   LMM: The call of "__fb_init" for structs

COMPATIBILITY_INFORMATION-OEM

The m4-export creates the __fb_init method stub in C now with a correct name and also creates a correct structure which is passed as argument.

The new structure contains a pInstance pointer, like FB_Inits for FBs do.

CDS-49376   Compile: reference assignment to reference

COMPATIBILITY_INFORMATION

Works for Compiler Version >= 3.5.10.0

If a property has a REFERENCE TO - type, then the code will behave differently from previous versions.

```
fb.prop := value;
```

will always call the get-Accessor and perform a write to the referenced value as returned by the get accessor

```
fb.__getprop()^ := value;
```

in previous versions the code was performed as set on the set-accessor, if the type was appropriate:

```
fb.__setprop(in := value);
```

with this call the reference was overwritten. There will now be a compatibility warning that informs about the change.

CDS-47295   LMM: Multiple assignments to interfaces variables

COMPATIBILITY_INFORMATION

With compilerversion >= 3.5.10.0 there is the following error message:

C0405: Multiple assignments to interface variables not allowed

CDS-47767   Compile: Generate Deprecated Warning for compiler versions < 3.4.0.0

COMPATIBILITY_INFORMATION

A compiler warning is generated if a project is compiled that uses a compiler version < 3.4.0.0.

CDS-49852 Store user passwords more safe, MD5 considered unsafe today

COMPATIBILITY_INFORMATION

The hashes for the user passwords can now be stored with a new format which supports salt and new hash algorithem. This improves the security of a password against guessing and brute-force. The password hash of a user is automatically convert with the first login after the setting was changed.

The protection against removing the user management and replacing password hashes with one from another project should be done by encrypting the project. Other mechanisms could be worked around or evaded after awhile.

The project looses its backward compatibility with the conversion to the new password hashes but you can convert the new password hashes back by disabling the setting and re-login of every user which logged in during the setting was enabled. The other option is to set a new password for every converted user when the setting is enabled.

CDS-50144 Device Application: Basic template for new structure

COMPATIBILITY_INFORMATION-OEM

A template for the device application structure has been added to CODESYS.

In order to allow users to utilize the full functionality of the new structure OEMs should:

consider using the static-area device description configuration setting for persistent and reatain data. This allows pre-allocating memory segments of a certain size for persistent variable lists per application and allows adding persistent variables during application development without the need to re-download the device application and all applications when adding persistent variables on application-level.

Include CmpLibrary in the list of required libraries in device descriptions to allow adding asynchronous calls to user POUs in the context of the device application bus cycle task.

Disable DeviceApplication support for specific targets if necessary (targets with strong memory constraints) setting the disable_device_application option in the runtime_features of the device description to false.

**CDS-49411 Setup: Install .Net 4.6 Framework**

GENERAL

.NET 4.6 is required for CODESYS

CDS-46519 Implement DeviceApplication: Changes to symbol configuration

COMPATIBILITY_INFORMATION

With a device application, the target setting ""symbolconfiguration\generate_as_separate_application"" must be FALSE. If it is set to TRUE, a compiler warning will be generated during symbol table generation, and the setting will be ignored.

It is not possible to add the Symbol Configuration directly to the Device Application. Use Symbol Configuration objects in normal applications to publish variables defined in the Device Application.

CDS-49093 Switch development system to .NET 4.6
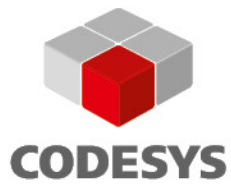
GENERAL

The executables of the setup solution (except DiagnosisToolLauncher.exe) have been upgraded to .Net Version 4.6.

COMPATIBILITY_INFORMATION-EndUser

In projects using the Target- or Webvisualization it might not be possible to login without an Online Change under the following condition: There is an imagepool referencing the same image more than once.

1.2 CODESYS Control

CDS-58510   CmpApp : Memory allocation / recycling on Application Download (possible "out of memory")

COMPATIBILITY_INFORMATION-EndUser

The download with online change uses an extra area to manage the online change. This area is now freed when it is no longer needed.

But this may not help if the allocation of areas defragments the available memory so that only (too) small blocks of memory are left. In this case a download of an application fails and a restart of the PLC is necessary.

CDS-50563   RTS: Release CmpSecurityManager

COMPATIBILITY_INFORMATION-OEM

The new component CmpSecurityManager is the central infrastructure to decide the level of all available security features in the runtime system!!

If the CmpSecurityManager is integrated in a runtime, the default level is to enable all security features as an option:

1. **CmpWebServer:**

If CmpSecurityManager is available in a runtime, the WebServer default cofiguration setting (see [CmpWebServer], ConnectionType) is overwritten by ""HTTP and HTTPS""!

If the WebServer is configured by the cfg-File of the runtime, this setting replaces the CmpSecurityManager default level!

2. **CmpApp:**

If CmpSecurityManager is available in a runtime, the application/bootproject can optionally be signed and encrypted.

3. **CmpSecureChannel:**

If CmpSecurityManager is available in a runtime, the CODESYS - RuntimeSystem communication can be used in plaintext and optional encrypted.

If you would like to have the same behaviour as before v3.5.10.0, you have to remove the CmpSecurityManager component and you have to set the compiler switch ""CMPSECURITYMANAGER_NOTIMPLEMENTED"".

CDS-25116   Gateway, RTS: Add additional fields (serial number etc.) to name service

COMPATIBILITY_INFORMATION-OEM

We have introduced the following limits for the strings transmitted as part of the network scan answer. This limits will in future also be used for other use cases of this strings. All the strings are returned by the SysTarget component of the runtime system, thus OEM specific implementations have to consider the following limits:

- Node name: 50 wide-char characters including NULL termination
- Device name: 100 wide-char characters including NULL termination (see remark below)
- Vendor name: 100 wide-char characters including NULL termination (see remark below)
- Serial number: 65 char characters including NULL termination

Remark: The length of the device name plus the length of the vendor name must not exceed 100 wide-char characters including NULL terminations.

It is recommended to implement the function SysTargetGetSerialNumber to return a human readable serial number of the device. A perfect solution is, to let the function return the same serial number, which is also printed on the PLC. So the user can relate the scan results to the physical PLCs based on the serial number.

Furthermore for OEMs it is possible to attach OEM speciifc data to the name service (network scan) answer. For this the CmpNameServiceService sends an event, which can be handled in an OEM runtime system component. This additonal information is transparently available on client side and is not interpreted by CODESYS products. The length of the OEM data is limited to 66 Bytes minus the length of the serial number (including NULL termination). See also description of the CmpNameServiceServer events.

COMPATIBILITY_INFORMATION-EndUser

The node name of the PLCs displayed in the CODESYS communication dialog is now limited to 50 wide-char characters including NULL termination. It is not possible anymore to assign a new PLC name longer than this limit.

If a runtime system of a PLC with a longer node name is updated to a version >= V3.5.10.0, then the node name will be cut to this limitation. If clients like DataServer, PLCHandler or OPC Server do a connect be the (old longer) node name, the connect will fail, because the PLC will answer with the shorter truncated name. In this case you have to update the connection parameters of the affected clients and/or rename the PLC.

CDS-10436   SymbolConfig: Access to IEC property variables of type struct and array must work

COMPATIBILITY_INFORMATION

The representation of properties in the Symbol XML file has changed. Starting with compiler version 3.5.7.0, property variables and members directly reference the corresponding type declaration, and a new attribute ""propertytype"" describes the type of the property. There is no separate artificial property type declared any more (which was difficult to parse, and did not contain all necessary information), thus property access via XML file should be completely transparent now.

KNOWN_LIMITATIONS

- Accessing properties of overly huge types (e. G. megabyte sized byte arrays) may lead to stack overflow (CDS-52128) - however, this will most likely also happen when calling the property directly from IEC code.
- Property access is affected by existing, not-yet fixed bugs in the IecVarAccess mechanism, one example is when the internal size is different from the external size (CDS-52126), which is much more likely to happen when properties are involved.

CDS-38300   CODESYS Control: Support TLS in SysLibSockets (SysSocket2)

COMPATIBILITY_INFORMATION-EndUser
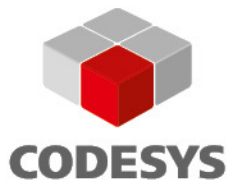
There are 2 new CODESYS libraries to support TLS sockets out of an IEC application:

- CmpTls.library
- SysSocket2.library

COMPATIBILITY_INFORMATION-OEM

There is a new interface function in the interface SysSocket2Itf in the runtime system:

```
STATICITF_DEF CLASSID CDECL SysSockGetClassId2(RTS_HANDLE hSocket, RTS_RESULT
*pResult);
```

This new interface must be implemented and supported in every OEM component which implements the SysSocket2Itf!

In SysSocket and CmpOpenSSL this implementation is still done. So in the Runtime Core everything is fine.

KNOWN_LIMITATIONS

On QNX runtime systems the SysSock2 functions in combination with TLS socket type won't work due to an unexpected system behavior (CDS-52447).

CDS-55681   IecVarAccess: The reading and writing of 64 bits Properties does not work

GENERAL

Access to properties of all basic types and all complex types containing no 64 bit basic type like (LWORD, ULINT, LTIME, LREAL, ...) is now also working on 32 bit CPUs. 64 bit CPUs did not show this issue.
But complex properties of type enum, array or struct or combination of these containing at least one 64 bit basic type do still not work and will be addressed later on by CDS-55810.

COMPATIBILITY_INFORMATION-EndUser

Compatibility note for 64 bit CPUs only (32 bit CPUs are not affected):
For the fix it was necessary to modify the CODESYS Control runtime system as well as the CODESYS SymbolicVarsBase.library.
This releation results in the following version constraint for writing (reading is not affected) properties of 64 bit basic types like (LWORD, ULINT, LTIME, LREAL, ...):
For CODESYS Control 64 bit runtime systems without this fix, it is important to use also a CODESYS compiler version and thereby also a SymbolicVarsBase.library version without this fix, e. g. the same version as the runtime system. The other direction - old compiler/library version without that fix in combination with a fixed runtime system will work. Ignoring this version constraint may result in wrong 64 bit basic type property values written by a caller of the Symbolconfiguration/IecVarAccess functionality.
Beside the IEC code, this functionality can also be used from OEM specific runtime system extension or by online clients like OPC Servers, Data Sources or PLCHandler-based clients.
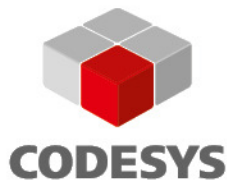
1.3 Driver

CDS-41297 Profinet Device Stack

KNOWN_LIMITATIONS

CDS-51830 Profinet Device (Certification): Check Local Port-Data

CDS-51834 Profinet Device (Certification): Change of IP fails on some plattforms(Windows, VxWorks, WinCE)

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 SP10 Patch 10"

fixVersion = "V3.5 SP10 Patch 10" AND issuetype = "New Feature"

List of features and changes since CODESYS V3.5 SP10:

fixVersion IN ("V3.5 SP10 Patch 10", "V3.5 SP10 Patch 9", "V3.5 SP10 Patch 8", "V3.5 SP10 Patch 7", "V3.5 SP10 Patch 6", "V3.5 SP10 Patch 5", "V3.5 SP10 Patch 4", "V3.5 SP10 Patch 3", "V3.5 SP10 Patch 2", "V3.5 SP10 Patch 1", "V3.5 SP10")

List of issues with compatibility information:

fixVersion IN ("V3.5 SP10 Patch 10", "V3.5 SP10 Patch 9", "V3.5 SP10 Patch 8", "V3.5 SP10 Patch 7", "V3.5 SP10 Patch 6", "V3.5 SP10 Patch 5", "V3.5 SP10 Patch 4", "V3.5 SP10 Patch 3", "V3.5 SP10 Patch 2", "V3.5 SP10 Patch 1", "V3.5 SP10") AND (text ~ GENERAL OR text ~ COMPATIBILITY_INFORMATION OR text ~ KNOWN_LIMITATIONS)

3 History

Created: Bernhard Reiterer (Quality Assurance)
Reviewed: Peter Zenker (Quality Assurance)
Released: Peter Zenker (Quality Assurance)