

There's more to it than that! New CODESYS features and products

CODESYS Users Conference 2016



Agenda

1

C integration

2

Arrays with variable lengths

3

Backup / Restore

4

OPC UA

5

Visualization

6

Fieldbus

7

Security

8

Miscellaneous

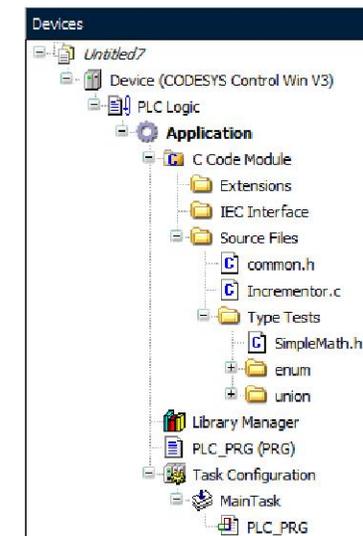
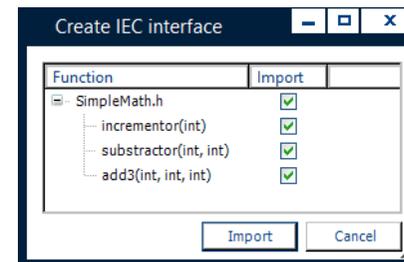
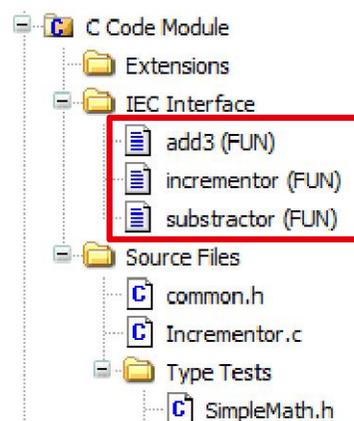
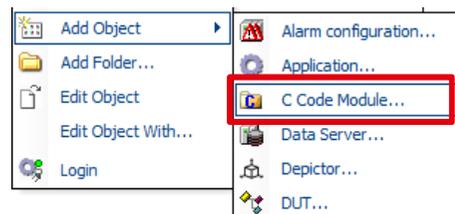


CODESYS C integration

- Add-on product for **device manufacturers**
- Integration of C objects in standard CODESYS projects
 - ➔ **End users** (and device manufacturers) **use C code**
- External toolchain for compiling/linking of C code in CODESYS plug-in component required
- Pre-compiled C code can be filed in libs (device manufacturers)
 - ➔ No external toolchain for users needed
 - ➔ Dynamic integration of c components
- Typical use cases
 - Re-usage of existing C code
 - Direct integration of generated C code
- CODESYS continues to be an IEC 61131-3 system!

CODESYS C integration - application

- New object type in object tree as of CODESYS V3.5 SP7
- Integration of a c folder structure
- Automatic generation of a suitable IEC 61131-3 interface from an imported C header file (*.h)



CODESYS C integration – application

- Easy access to generated functions within the IEC 61131-3 code
- C code referencing in generated interface POU's
- C code execution by calling the interface POU's

```

1  add3
2  {attribute 'C_SOURCE_EXPORT' := 'add3'}
3  {attribute 'external_name' := 'add3_CExt'}
4  FUNCTION add3 : DINT
5  VAR_INPUT
6     v1 : DINT;
7     v2 : DINT;
8     v3 : DINT;
9  END_VAR
10 VAR
11 END_VAR
12
13
14
15
16
17
18
19
20
21
22
23

```

```

1  Incrementor.c
2  #include "common.h"
3
4  /* Example for a external library functions */
5  int incrementor(int val)
6  {
7     return val + 77;
8  }
9
10 int subtractor(int v1, int v2)
11 {
12     return v1 - v2;
13 }
14
15 int add3(int v1, int v2, int v3)
16 {
17     return v1 + v2 + v3;
18 }
19
20 static int hiddenAdd3(int v1, int v2, int v3)
21 {
22     return v1 + v2 + v3;
23 }

```

```

1  PLC_PRG
2  PROGRAM PLC_PRG
3  VAR
4     result: DINT;
5  END_VAR
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23

```

```

1  result:=add3(v1:= 2, v2:= 3, v3:= 5);
2

```



Agenda

1

C integration

2

Arrays with variable lengths

3

Backup / Restore

4

OPC UA

5

Visualization

6

Fieldbus

7

Security

8

Miscellaneous

Arrays with variable lengths

Rather: Array transfer with variable lengths

- Functions (FUN)/function blocks (FB) with arrays of variable length as input:
 - Conformal to IEC 61131-3, 3rd Edition

```

FUNCTION SUM: DINT;           // Function that adds array numbers of variable length
VAR IN OUT
  A: ARRAY [*] OF INT;      // Array is transferred without fixed limits
END_VAR
VAR
  i, sum2 : DINT;
  LB: DINT;                 // Variable for lower array limit
  UB: DINT;                 // Variable for upper array limit
END_VAR

sum2:= 0;                   // Reset intermediate total
LB:=LOWER_BOUND(A,1);      // Calculate array limits
UB:=UPPER_BOUND(A,1);

FOR i:= LB TO UB DO        // Loop within array limits
  sum2:= sum2 + A[i];      // Calculate intermediate total
END_FOR;
SUM:= sum2;                // Transfer final result

arArray1: ARRAY[0..5] OF INT := [1, 2, 3, 4, 5, 6]; //Array with 5 fields
arArray2: ARRAY[1..4] OF INT := [7, 6, 5, 4];      //Array with 4 fields

// Multiple call of a function with arrays of different length
Result:=SUM(A:= arArray1);
Result2:=SUM(A:= arArray2);
  
```



Agenda

1

C integration

2

Arrays with variable lengths

3

Backup / Restore

4

OPC UA

5

Visualization

6

Fieldbus

7

Security

8

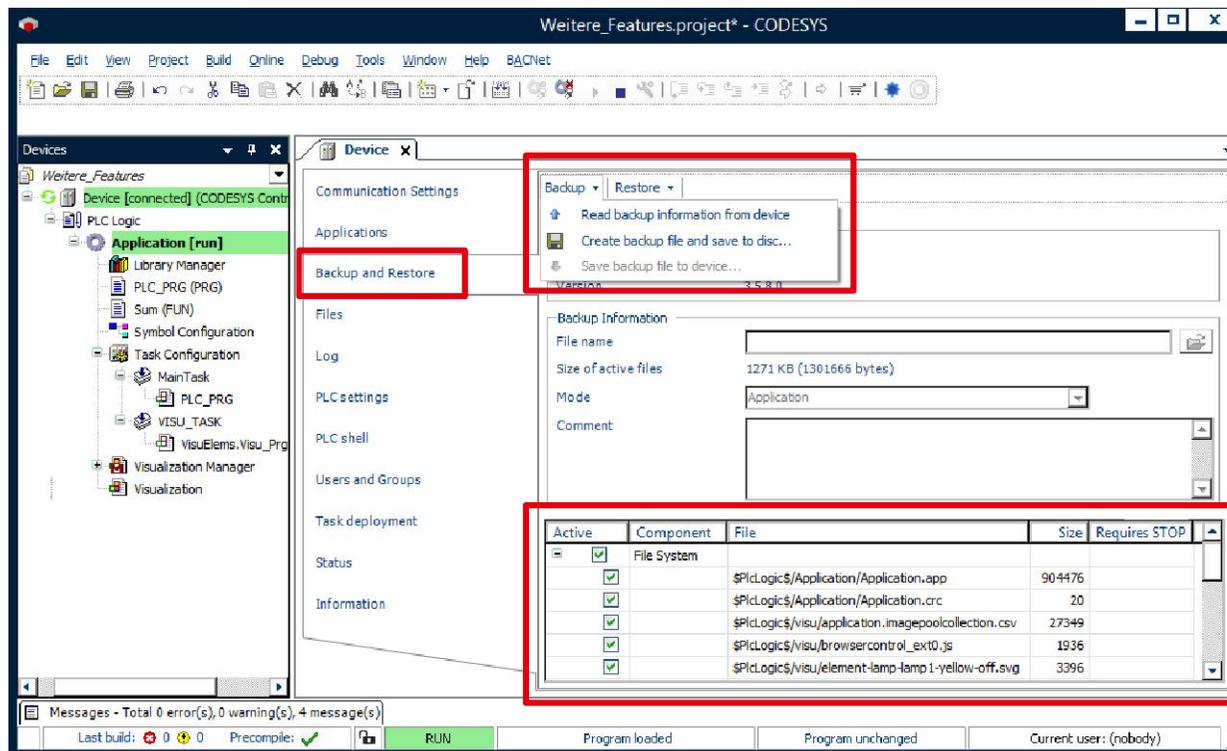
Miscellaneous



What if the controller breaks down during operation?

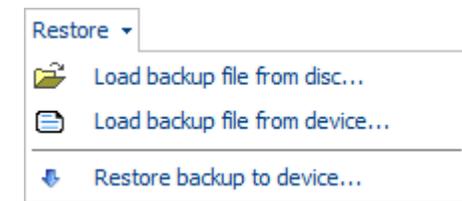
- Use a new controller!
- Load the application anew – but:
 - Which one? Which CODESYS version?
 - Which settings?
 - What about remanent data?
- **Solution:** Backup / Restore for a convenient disaster recovery
- **Prerequisite:**
 - PLC using CODESYS Control V3.5 SP8
 - Boot application has been generated
 - Backup has been made in time and is now available
 - Identical controller available in exchange
- **Benefits:**
 - No need for a repeated compilation, project remains the same, remanent data

Disaster recovery: Procedure



The screenshot shows the CODESYS interface with the 'Backup and Restore' menu open. The 'Backup and Restore' option in the left sidebar is highlighted with a red box. The 'Backup and Restore' sub-menu is also highlighted with a red box, showing options: 'Read backup information from device', 'Create backup file and save to disc...', and 'Save backup file to device...'. Below this, a table lists active components and files for backup.

Active	Component	File	Size	Requires STOP
<input checked="" type="checkbox"/>	File System			
<input checked="" type="checkbox"/>		\$PldLogic\$/Application/Application.app	904476	
<input checked="" type="checkbox"/>		\$PldLogic\$/Application/Application.crc	20	
<input checked="" type="checkbox"/>		\$PldLogic\$/visu/application.imagepoolcollection.csv	27349	
<input checked="" type="checkbox"/>		\$PldLogic\$/visu/browsercontrol_ext0.js	1936	
<input checked="" type="checkbox"/>		\$PldLogic\$/visu/element-lamp-lamp1-yellow-off.svg	3396	



The 'Restore' menu is shown with the following options:

- Load backup file from disc...
- Load backup file from device...
- Restore backup to device...



Disaster recovery: Recommendations and outlook

- Recommendations:
 - Use current PLC
 - Backup after completion of commissioning or maintenance at the latest
 - Provide a suitable folder structure for backups
- In the pipeline for development:
 - Automatic backup for “Create boot application”
 - Backup through application → data backup for disaster recovery
 - “Logical” filing location for backup data
- More on this topic at the CODESYS Users Conference 2017...



Agenda

1

C integration

2

Arrays with variable lengths

3

Backup / Restore

4

OPC UA

5

Visualization

6

Fieldbus

7

Security

8

Miscellaneous

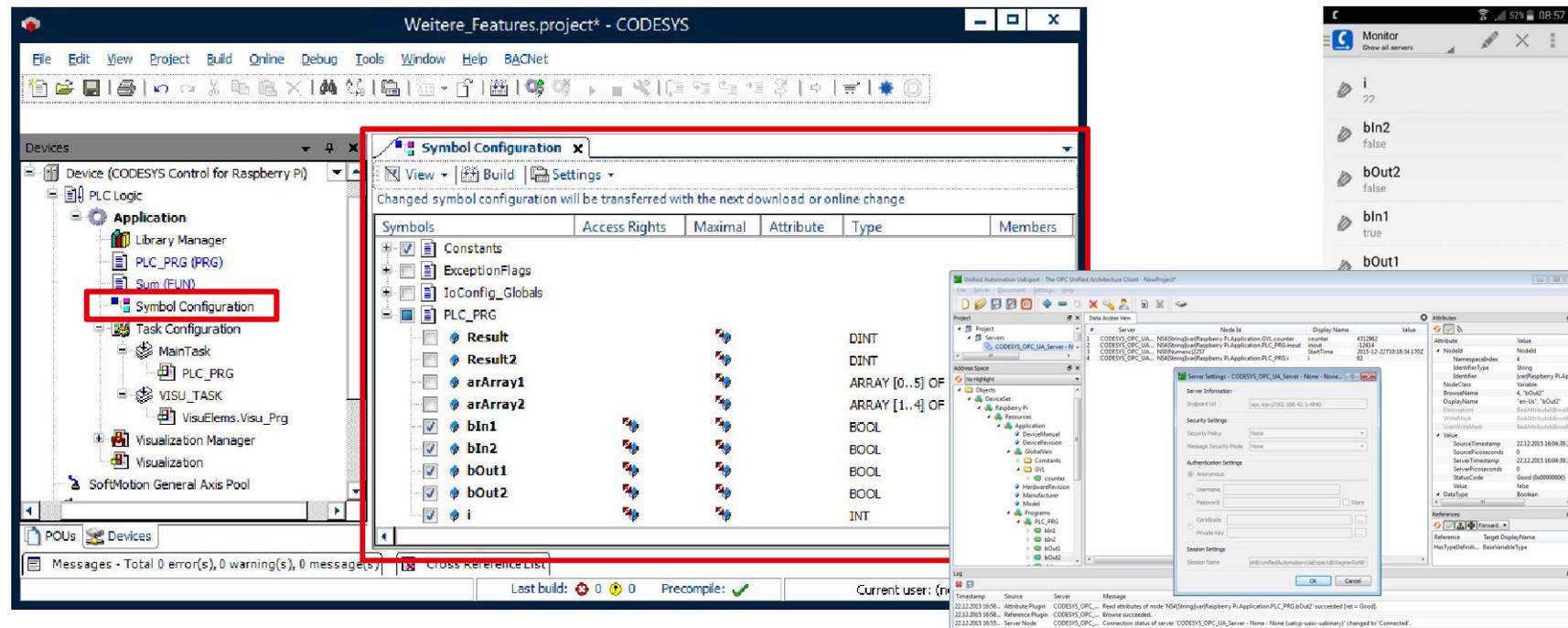


CODESYS OPC UA Server

- Ethernet-based data exchange for all system platforms
- Standard for Industry 4.0 / Industrial Internet
- CODESYS OPC UA
 - Integrated in the CODESYS Control runtime system (SoftPLC on the device)
 - ➔ Portable to (nearly) any platform
 - System requirement: RAM / Flash ca. 1 MB
 - Implementation by device manufacturer
 - Included in all SoftPLC systems in the CODESYS Store!
- **Outlook:**
 - Encrypted communication and user administration
 - OPC UA Client and PLCopen POU's

CODESYS OPC UA Server - application

- Export of relevant data via symbol configuration
- Download onto the controller with integrated CODESYS OPC UA server
- Client: Establish connection to the server



The image displays three overlapping windows from the CODESYS environment:

- Symbol Configuration:** A table showing the configuration of symbols to be exported. The table has columns for Symbols, Access Rights, Maximal, Attribute, Type, and Members. The symbols listed include constants, exception flags, PLC_PRG, and various data types like Result, arrays, and booleans.
- Server Settings:** A dialog box for configuring the OPC UA server. It includes sections for Server Information, Security Settings, Authentication Settings, and Session Settings. The server name is set to 'CODESYS OPC UA Server - None - None'.
- Monitor:** A mobile application interface showing the status of the server. It lists variables such as 'i', 'bIn2', 'bOut2', 'bIn1', and 'bOut1' with their current values (e.g., false, true).



Agenda

1

C integration

2

Arrays with variable lengths

3

Backup / Restore

4

OPC UA

5

Visualization

6

Fieldbus

7

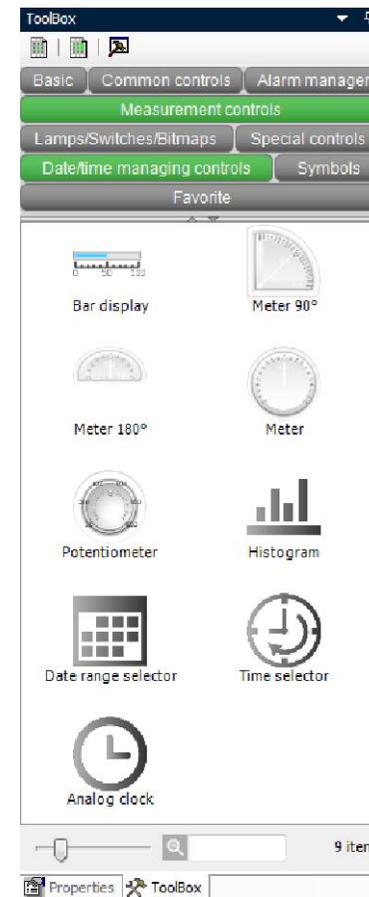
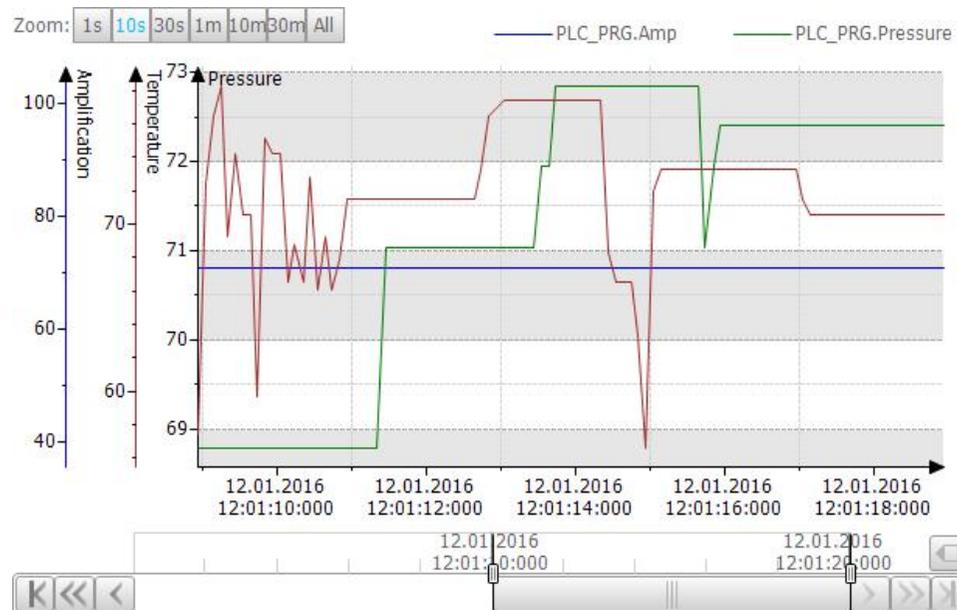
Security

8

Miscellaneous

What we have not yet presented in detail (1/2)

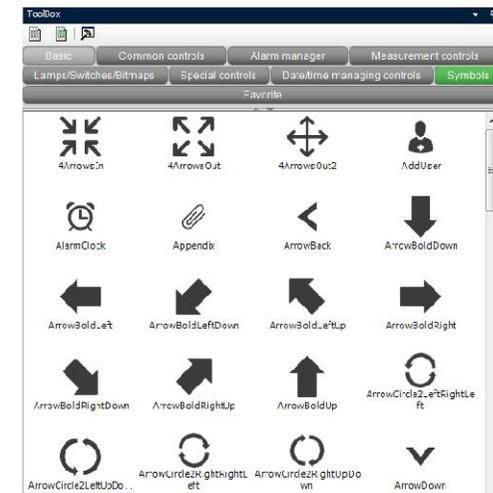
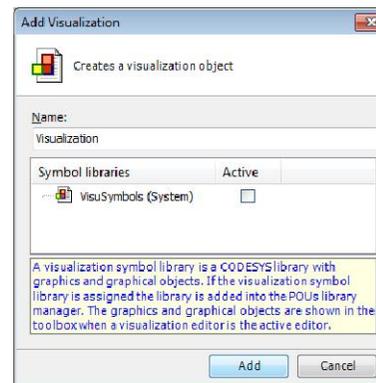
- Improved trend function: now with multiple Y-axes
- New toolbox



What we have not yet presented in detail (2/2)

- New symbol library including 200 scalable vector graphics (SVG)
 - Symbol library: CODESYS Lib with additional features
 - Easy integration of own symbol libraries
- New visualization element: analog clock

Property	Value
Elementname	GenElemInst_13
Type of element	Analog clock
Position	
X	481
Y	29
Width	165
Height	166
Display time	
Use system time	<input type="checkbox"/>
Variable	PLC_PRG.Now
Design	From style
Scaling type	Anisotropic
Optimized drawing	<input checked="" type="checkbox"/>





Agenda

1

C integration

2

Arrays with variable lengths

3

Backup / Restore

4

OPC UA

5

Visualization

6

Fieldbus

7

Security

8

Miscellaneous

New products (configuration & protocol stacks)

- Controller with vacant Ethernet port now extensible with EtherNet/IP:

- CODESYS EtherNet/IP Scanner (Master) SL

- Controller turns to EtherNet/IP „Master“

- CODESYS EtherNet/IP Adapter (Slave) SL

- Controller turns to EtherNet/IP „Device“

- CODESYS controller within another controller network



- EtherCAT Gateway modules supported now:

- CODESYS PROFIBUS Master (for EL6731) SL

- CODESYS PROFIBUS Slave (for EL6731-0010) SL

- CODESYS PROFINET Controller (for EL6631) SL

- CODESYS PROFINET Device (for EL6631-0010) SL



- Directly integrate/configure PROFIBUS/PROFINET in the EtherCAT network

- Prerequisite for all products: Single device license on the device



Agenda

1

C integration

2

Arrays with variable lengths

3

Backup / Restore

4

OPC UA

5

Visualization

6

Fieldbus

7

Security

8

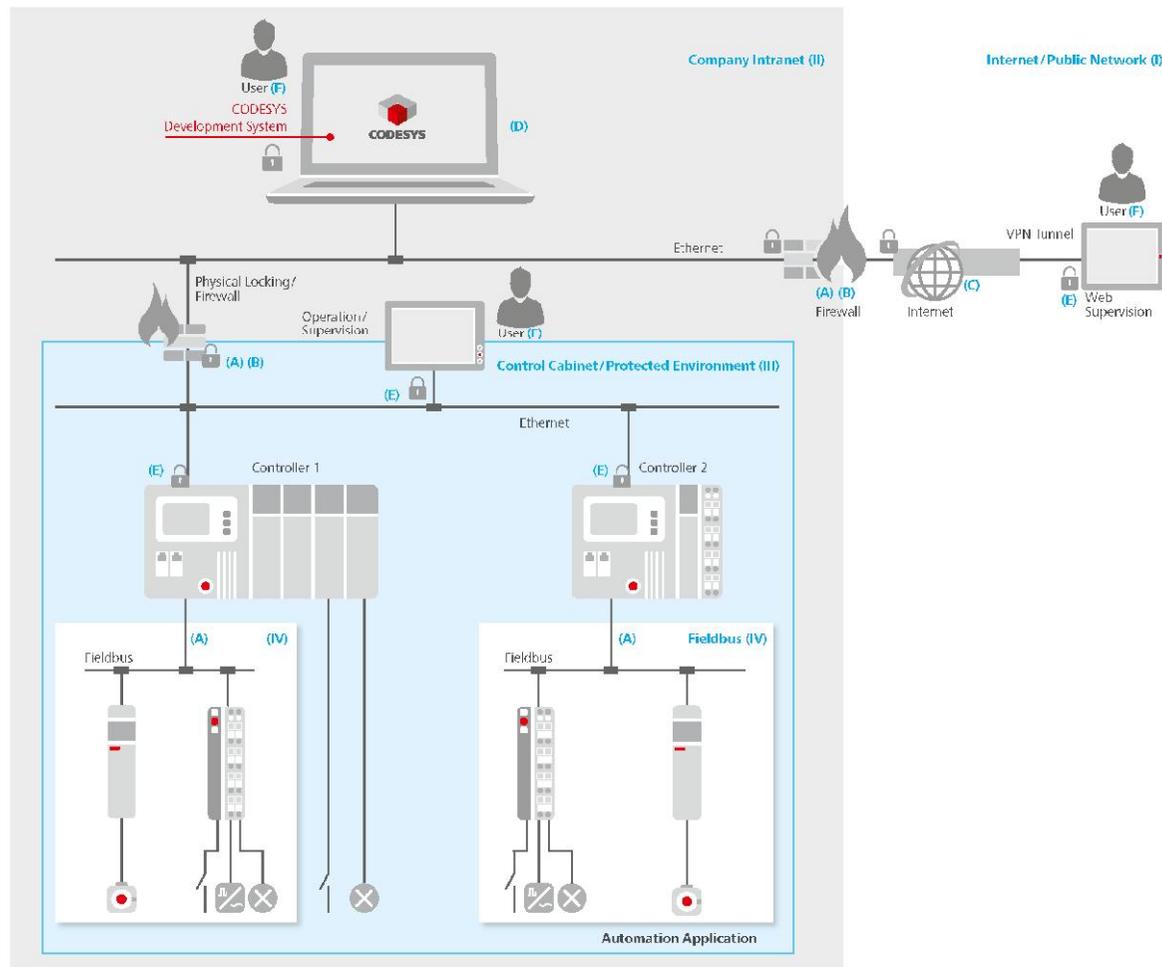
Miscellaneous



Security whitepaper available – Instructions for information security

- Contents:
 - Security in industrial control applications
 - General measures/recommendations for industrial plants
 - Responsibilities
 - Overview of available/future features in CODESYS to improve security
- Can be requested at security@codesys.com
- Only available in English language

Excerpt from the whitepaper: Protection areas in automation environments





Agenda

1

C integration

2

Arrays with variable lengths

3

Backup / Restore

4

OPC UA

5

Visualization

6

Fieldbus

7

Security

8

Miscellaneous



Many small improvements / bug fixes

- Quality effort in CODESYS V3.5 SP8:
Altogether > 1,500 improvements
- Improvements in all CODESYS product areas (CODESYS Development System, Visualization, Runtime System etc.)
- Improvements in all severity levels
- **Strategy:**
Last in, first out → Fast debugging of reported errors
- **Goal:**
 - Fix all errors reported recently within a narrow time frame
 - Fix all errors users have long been waiting for



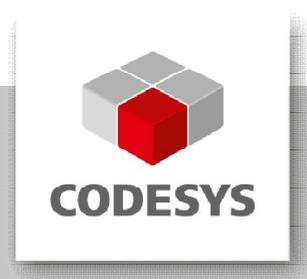
No standstill!

- CODESYS is permanently developed further!
- Focus not only on new functionality but also on improvements
- Overview of the major improvements available for each main version and Service Packs

And finally:

Your eyes are not quite so good any more?

CODESYS now with 120 dpi resolution!



Inspiring Automation Solutions

Thank you for your attention.

CODESYS® is a registered trademark of 3S-Smart Software Solutions GmbH. Technical specifications are subject to change. Errors and omissions excepted. No reproduction or distribution, in whole or in part, without prior permission.