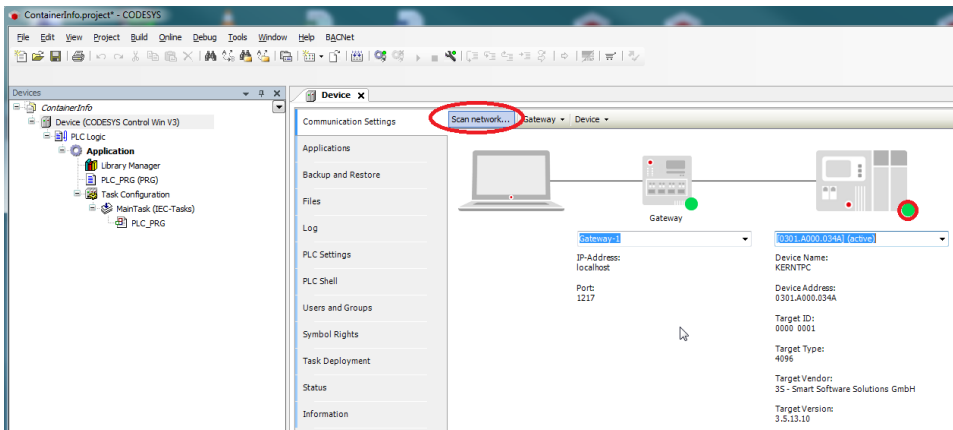
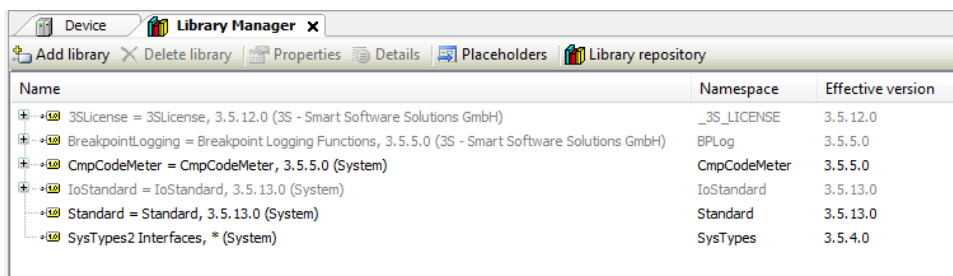


Licensing: Querying the Container Information

- Create a "Standard project" and select *CODESYS Control Win V3* as the device.
- Define the target system by means of the *Network scan*.



- Open the *Library Manager* and add the following libraries:
CmpCodeMeter
SysTypes2 interfaces



- Create a "Structure" named *ST_CODEMETER_INFO_EX*.
Adapt the *ST_CODEMETER_INFO_EX* as follows:

Declaration

```
TYPE ST_CODEMETER_INFO_EX :  
  STRUCT  
    stRTS_CODEMETER_INFO : CmpCodeMeter.RTS_CODEMETER_INFO;  
    wsContainerName : WSTRING;  
  END_STRUCT  
END_TYPE
```

- Adapt the POU *PLC_PRG* as follows:

Declaration

```

VAR CONSTANT
    cw_MaxInfos      : WORD := 3;
END_VAR
VAR
    xReadInfo        : BOOL;
    iecResult         : SysTypes.RTS_IEC_RESULT;
    iecHandle         : SysTypes.RTS_IEC_HANDLE;
    iecPrevHandle     : SysTypes.RTS_IEC_HANDLE;
    wIndex            : WORD;
    aInfo             : ARRAY [0..cw_MaxInfos] OF ST_CODEMETER_INFO_EX;
END_VAR

```

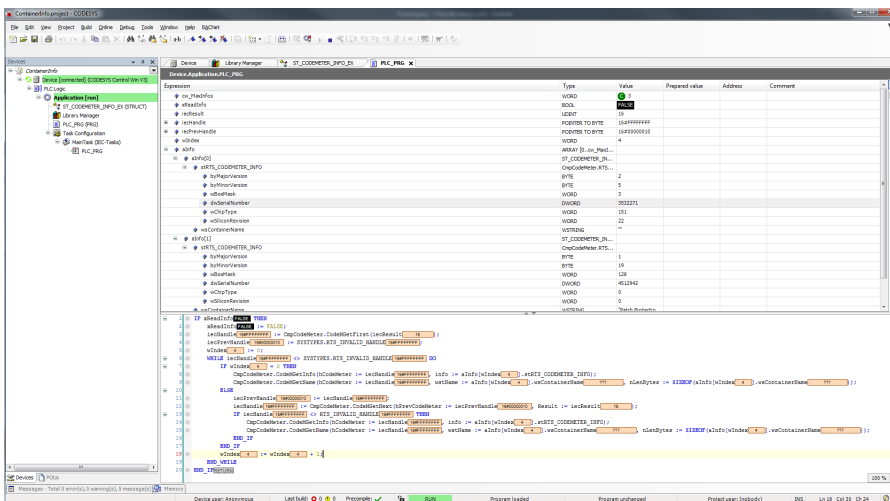
Implemen tation

```

IF xReadInfo THEN
    xReadInfo := FALSE;
    iecHandle := CmpCodeMeter.CodeMGetFirst(iecResult);
    iecPrevHandle := SYSTYPES.RTS_INVALID_HANDLE;
    wIndex := 0;
    WHILE iecHandle <> SYSTYPES.RTS_INVALID_HANDLE DO
        IF wIndex = 0 THEN
            CmpCodeMeter.CodeMGetInfo(hCodeMeter := iecHandle, info := aInfo[wIndex].
stRTS_CODEMETER_INFO);
            CmpCodeMeter.CodeMGetName(hCodeMeter := iecHandle, wstName := aInfo[wIndex].
wsContainerName, nLenBytes := SIZEOF(aInfo[wIndex].wsContainerName));
        ELSE
            iecPrevHandle := iecHandle;
            iecHandle := CmpCodeMeter.CodeMGetNext(hPrevCodeMeter := iecPrevHandle, Result :=
iecResult);
        IF iecHandle <> RTS_INVALID_HANDLE THEN
            CmpCodeMeter.CodeMGetInfo(hCodeMeter := iecHandle, info := aInfo[wIndex].
stRTS_CODEMETER_INFO);
            CmpCodeMeter.CodeMGetName(hCodeMeter := iecHandle, wstName := aInfo[wIndex].
wsContainerName, nLenBytes := SIZEOF(aInfo[wIndex].wsContainerName));
        END_IF
    END_IF
    wIndex := wIndex + 1;
END_WHILE
END_IF

```

- Start the project and set the *xReadInfo* variable to *TRUE*.



The read container/dongle information is saved in the array.

For comparing the information from the *License Manager*.

The screenshot displays a C program in a code editor and a License Manager window. The code reads RTSS dongle information and compares it with a license manager. The License Manager window shows a list of products and their licenses, with the 'Local OnDisk [3-353227]' container selected.

Code Editor:

```
1  IF xReadInfo[0] THEN
2  xReadInfo[0] := FALSE;
3  iecHandle[0] := CmpCodeMeter.CodeMeterFirst(iecResult[0]);
4  iecPrevHandle[0] := SYSTYPES.RTS_INVALID_HANDLE[0];
5  wIndex[0] := 0;
6  WHILE iecHandle[0] <> SYSTYPES.RTS_INVALID_HANDLE[0] DO
7  IF wIndex[0] = 0 THEN
8  CmpCodeMeter.CodeMeterInfo(hCodeMeter := iecHandle[0], info := aInfo[wIndex[0]]);
9  CmpCodeMeter.CodeMeterName(hCodeMeter := iecHandle[0], wName := aInfo[wIndex[0]]);
10 ELSE
11 iecPrevHandle[0] := iecHandle[0];
12 iecHandle[0] := CmpCodeMeter.CodeMeterNext(hPrevCodeMeter := iecPrevHandle[0]);
13 IF iecHandle[0] <> SYSTYPES.RTS_INVALID_HANDLE[0] THEN
14 CmpCodeMeter.CodeMeterInfo(hCodeMeter := iecHandle[0], info := aInfo[wIndex[0]]);
15 CmpCodeMeter.CodeMeterName(hCodeMeter := iecHandle[0], wName := aInfo[wIndex[0]]);
16 END IF;
17 wIndex[0] := wIndex[0] + 1;
18 END WHILE;
19 END_IF;
20 END_PROGRAM;
```

License Manager:

Container: Local OnDisk [3-353227]

Products:

- CODESYS SVN
- CODESYS Test Manager
- CODESYS Control Win32
- CODESYS Vss Element Toolkit
- CODESYS Static Analysis
- CODESYS Application Composer
- CODESYS BACnet SL
- CODESYS Control for Raspberry Pi
- IOV-EL624
- CSV Library
- CODESYS UML
- SNMP Library
- JSON HTTP Client Library
- JSON UMLs
- MathService_1.0.0.0

Name: CODESYS SVN
Company: 3S- Smart Software Solutions GmbH
Unit counter: -
License quantity: -
Usage period: -
Feature map: -
Activation time: -
Expiration time: -
Firm code: 101597
Product code: 8747
Description: A Subversion source control integration

The license for the product exists on the selected license container.
The license for the product exists on the selected license container, but is invalid (e.g. expired).
The license for the product does not exist on the selected license container.