BACnet: Reading and writing a Calendar object

The BACnet server from the FAQ *BACnet: Cyclic Reading of a Property* is used as the server here.

Requirements for the BACnet server

- If you have followed the steps in the above linked FAQ exactly, then
  - add a BACnet object of type BACnet Calendar to the BACnet Server in the device tree
  - delete the exisiting BACnet Analog Value
- Adapt the "PLC_PRG" as follows:

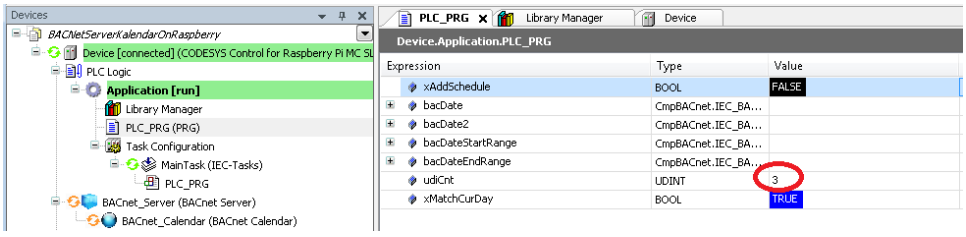
Declaration

```
VAR
        xAddCalendar                : BOOL;
        // API does not check dayOfweek!!! == >> DAY_OF_WEEK_UNSPECIFIED
        bacDate                                 : CmpBACnet.IEC_BACNET_DATE := (year := 2020,
month := 2, dayOfMonth := 7, dayOfWeek := CmpBACnet.IEC_BACNET_DAY_OF_WEEK.DAY_OF_WEEK_FRIDAY);
        bacDate2                        : CmpBACnet.IEC_BACNET_DATE := (year := 2020, month := 2,
dayOfMonth := 14, dayOfWeek := CmpBACnet.IEC_BACNET_DAY_OF_WEEK.DAY_OF_WEEK_UNSPECIFIED);
        bacDateStartRange       : CmpBACnet.IEC_BACNET_DATE := (year := 2020, month := 2,
dayOfMonth := 17, dayOfWeek := CmpBACnet.IEC_BACNET_DAY_OF_WEEK.DAY_OF_WEEK_UNSPECIFIED);
        bacDateEndRange                 : CmpBACnet.IEC_BACNET_DATE := (year := 2020, month := 3,
dayOfMonth := 25, dayOfWeek := CmpBACnet.IEC_BACNET_DAY_OF_WEEK.DAY_OF_WEEK_UNSPECIFIED);

        udiCnt                                  : UDINT;
        xMatchCurDay                : BOOL;
END_VAR
```

Implemen
tation

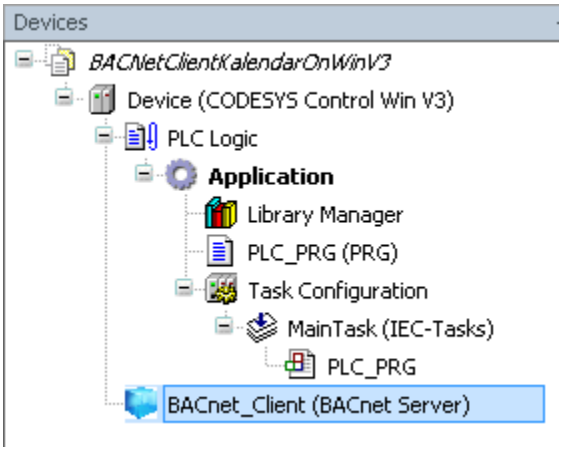
```
BACnet_Calendar.GetEntryCount(count => udiCnt);
xMatchCurDay := BACnet_Calendar.PresentValue;
IF xAddCalendar THEN
        xAddCalendar := FALSE;
        BACnet_Calendar.AddBACnetDate(bacDat := bacDate);
        BACnet_Calendar.AddBACnetDate(bacDat := bacDate2);
        BACnet_Calendar.AddBACnetDateRange(bacStartDat := bacDateStartRange, bacEndDat :=
bacDateEndRange);
END_IF
```

- Download the project to the controller and set the *xAddCalendar* variable to *TRUE*.
  If successful, the entry count for the BACnet Calendar will increase by three.

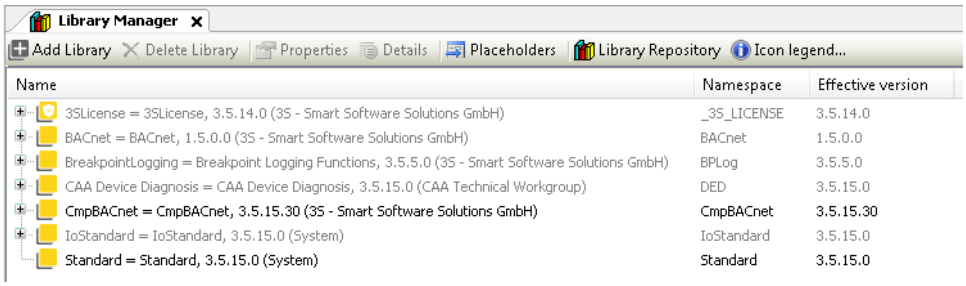


Requirements for the BACnet client

- Create a "Standard project" and select *CODESYS Control Win V3* as the device.
- Define the target system by means of the network scan (see BACnet server).
- Insert a "BACnet Server" object and rename it to "BACnet_Client"



- Open the *Library Manager* and add the following libraries *CmpBACnet*

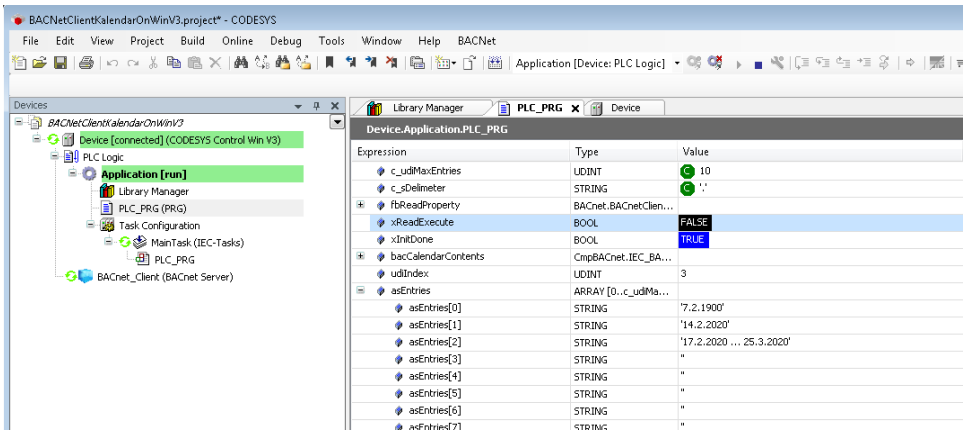


- Adapt the "PLC_PRG" as follows:

Declaration

```
VAR CONSTANT
        c_udiMaxEntries               : UDINT := 10;
        c_sDelimeter                : String := '.';
END_VAR
VAR
        fbReadProperty                : BACnet.BACnetClientReadProperty;
        xReadExecute                : BOOL;

        xInitDone                        : BOOL := FALSE;
        bacCalendarContents        : CmpBACnet.IEC_BACNET_PROPERTY_CONTENTS;
        udiIndex                        : UDINT;

        asEntries                        : ARRAY [0..c_udiMaxEntries] OF STRING;

        pbyRawBuffer                : POINTER TO BYTE;
        pCalenderEntry                : POINTER TO CmpBACnet.IEC_BACNET_CALENDAR_ENTRY;
        dat                : CmpBACnet.IEC_BACNET_DATE;
        // Value is used if type is set to "date range"
        dateRange           : CmpBACnet.IEC_BACNET_DATE_RANGE;
        // Value is used if type is set to "week-n-day"
        weekNDay            : CmpBACnet.IEC_BACNET_WEEK_AND_DAY;

        sHelp                                : STRING;
END_VAR
```

Implemen
tation

```
IF NOT xInitDone THEN
        fbReadProperty.RegisterToServer(BACnet_Client);
        fbReadProperty(dwTargetDeviceNumber := 718,
                                    objType := BACnet.CmpBACnet.IEC_BACNET_OBJECT_TYPE.
OBJ_CALENDAR, objInst := 1,
                                    propID := BACnet.CmpBACnet.IEC_BACNET_PROPERTY_ID.
PROP_DATELIST,
                                        udiTimeOut := 5000000);
        xInitDone := TRUE;
ELSE
        fbReadProperty(xExecute := xReadExecute);
        IF fbReadProperty.xDone THEN
                xReadExecute := FALSE;
                bacCalendarContents := fbReadProperty.result;

                IF bacCalendarContents.tag = CmpBACnet.IEC_BACNET_DATA_TYPE.
DATA_TYPE_CALENDAR_ENTRY THEN
                        pCalenderEntry := bacCalendarContents.buffer.pBuffer;
                        FOR udiIndex := 0 TO bacCalendarContents.nElements -1 DO
                                CASE pCalenderEntry[udiIndex].tag OF
                                        CmpBACnet.IEC_BACNET_CALENDAR_ENTRY_TYPE.
CALENDAR_ENTRY_DATE:
                                                sHelp := CONCAT(BYTE_TO_STRING(pCalenderEntry
[udiIndex].entry.dat.dayOfMonth), '.');
                                                sHelp := CONCAT(sHelp, TO_STRING(pCalenderEntry
[udiIndex].entry.dat.month));
                                                sHelp := CONCAT(sHelp, '.');
                                                sHelp := CONCAT(sHelp, WORD_TO_STRING
(pCalenderEntry[udiIndex].entry.dat.year));
                                                asEntries[udiIndex] := sHelp;
                                                CmpBACnet.IEC_BACNET_CALENDAR_ENTRY_TYPE.
CALENDAR_ENTRY_DATE_RANGE:
                                                sHelp := CONCAT(BYTE_TO_STRING(pCalenderEntry
[udiIndex].entry.dateRange.startDate.dayOfMonth), '.');
                                                sHelp := CONCAT(sHelp, TO_STRING(pCalenderEntry
[udiIndex].entry.dateRange.startDate.month));
                                                sHelp := CONCAT(sHelp, '.');
                                                sHelp := CONCAT(sHelp, WORD_TO_STRING
(pCalenderEntry[udiIndex].entry.dateRange.startDate.year));
                                                sHelp := CONCAT(sHelp, ' ... ');
                                                sHelp := CONCAT(sHelp, BYTE_TO_STRING
(pCalenderEntry[udiIndex].entry.dateRange.endDate.dayOfMonth));
                                                sHelp := CONCAT(sHelp, '.');
                                                sHelp := CONCAT(sHelp, TO_STRING(pCalenderEntry
[udiIndex].entry.dateRange.endDate.month));
                                                sHelp := CONCAT(sHelp, '.');
                                                sHelp := CONCAT(sHelp, WORD_TO_STRING
(pCalenderEntry[udiIndex].entry.dateRange.endDate.year));
                                                asEntries[udiIndex] := sHelp;
                                        CmpBACnet.IEC_BACNET_CALENDAR_ENTRY_TYPE.
CALENDAR_ENTRY_WEEK_AND_DAY:
                                                // Do Something
                                        ELSE
                                                // Should never used == >> Implement ErrorHandling
                                END_CASE
                                IF pCalenderEntry[udiIndex].tag = CmpBACnet.
IEC_BACNET_CALENDAR_ENTRY_TYPE.CALENDAR_ENTRY_DATE THEN
                                        dat := pCalenderEntry[udiIndex].entry.dat;
                                END_IF
                                IF pCalenderEntry[udiIndex].tag = CmpBACnet.
IEC_BACNET_CALENDAR_ENTRY_TYPE.CALENDAR_ENTRY_DATE_RANGE THEN
                                        dateRange := pCalenderEntry[udiIndex].entry.dateRange;
                                END_IF
                        END_FOR
                END_IF
        END_IF
END_IF
```

- Download the project to the controller and set the *xReadExecute* auf den Wert *TRUE*.
  If successful, the values are transmitted from the server to the client.