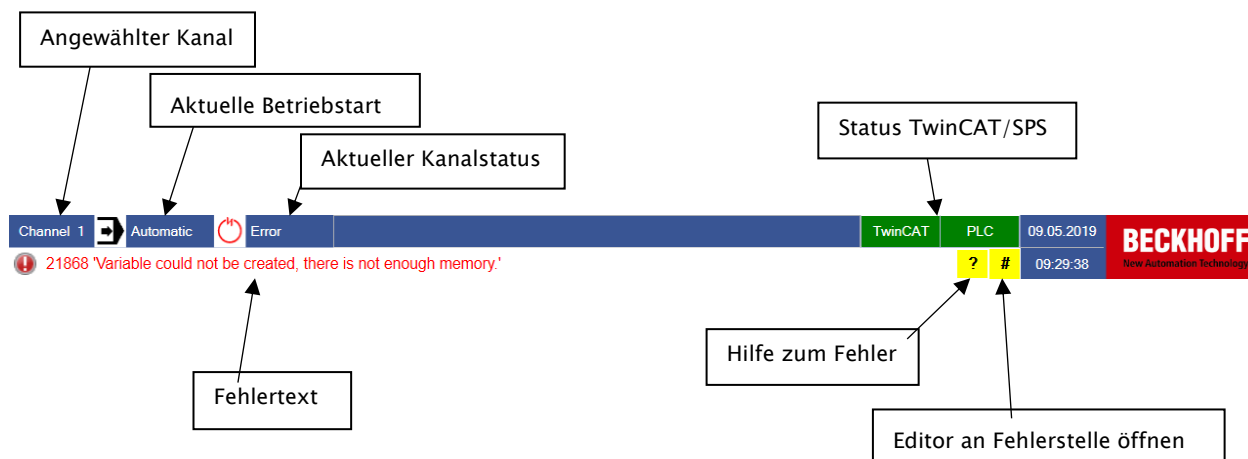


Letzte Änderung: 10.9.2020

# Header

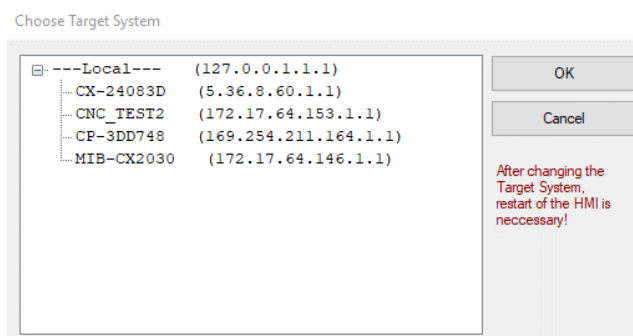
Falls das Interface „Beckhoff.App.Core.Interfaces.IBAHeader“ in [Unity](#) definiert ist, wird diese Implementierung zur Anzeige des Headers genutzt und alle älteren Versionen werden ausgeblendet. Somit besteht die Möglichkeit, über diesen Weg eine eigene Implementierung eines Headers zu realisieren.

Im Standard wird „IBAHeader“ durch „Beckhoff.App.Header.HeaderCtrl“ aus der DLL „Beckhoff.App.Header.dll“ implementiert, was in der Unity Konfigurationsdatei „Header.config.xml“ im Ordner „System\locConfig“ definiert ist.



Doppelklick auf die Fehlerzeile öffnet die ausführliche Fehleranzeige.

Klick auf „Status TwinCAT/SPS“ öffnet ein Fenster, in dem das Zielsystem ausgewählt werden kann:



Klick auf das Fragezeichen öffnet die ausführliche Fehlerdokumentation zu dem Fehler  
Klick auf „#“ öffnet den NC Editor an der Stelle im NC Programm, wo der Fehler auftritt.

In den Settings gibt es Einstellungen für den Header im Abschnitt „MessageLine“:

- „CNC DignosisManual English/German“ beinhalten den Ort der .chm Hilfedateien der entsprechenden Sprache.
- „EditButtonVisible“: ist dieser ausgeschaltet, wird der Knopf zum Editieren „#“ nie eingeblendet.
- „Font“: Schriftart und Größe der Fehlerzeile.
- „FormToShowOnDoubleClick“: Instanzname der Form die bei Doppelklick angezeigt wird.
- „ShowOnlyHighestStatus“: Es werden nur die „wichtigsten“ Meldungen angezeigt. Gibt es zum Beispiel Fehler und Warnungen werden bei „TRUE“ nur Fehler angezeigt.
- „TimerIntervall“: Intervall mit dem bei mehreren anstehenden Meldungen die jeweils nächste Meldung angezeigt wird.

## Interface IBAHeader

Das Interface IBAHeader ist folgendermaßen definiert:

```

namespace Beckhoff.App.Core.Interfaces
{
    public interface IBAHeader
    {
        Control HeaderControl { get; }
        /// <summary>
        /// additional Control, only displayed if != null
        /// </summary>
        void AddControl(System.Windows.Controls.UserControl control);
        /// <summary>
        /// additional Control, only displayed if != null
        /// </summary>
        void RemoveControl(System.Windows.Controls.UserControl control);
    }
}

```

Die beiden Methoden „AddControl“ bzw. „RemoveControl“ ermöglichen es, zusätzliche WPF UserControls dem Header zur Laufzeit hinzuzufügen bzw. zu entfernen.

Über Unity kann IBAHeader abgefragt und dann benutzt werden.

Beispiel:

```

public FormDemo(IBAHeader header, string instanceName) : this()
{
    header = header;
    headerVM = new HeaderControlVM(instanceName);
    _headerControl = new ucHeaderControl {DataContext = _headerVM};
    header.AddControl( headerControl);
}

```



**Beckhoff Automation**  
 GmbH & Co. KG  
 Hülshorstweg 20  
 33415 Verl, Germany

Postfach 11 42  
 33398 Verl

Telefon: +49 5246 963-0  
 Vertrieb: +49 5246 963-1000  
 Service: +49 5246 963-460  
 Support: +49 5246 963-157

Fax Zentrale: - 149  
 Fax Vertrieb: - 198

www.beckhoff.de  
 info@beckhoff.de  
 vertrieb@beckhoff.de  
 service@beckhoff.de  
 support@beckhoff.de  
 international-sales@beckhoff.de  
 engineering@beckhoff.de

Geschäftsführer:  
 Dipl.-Phys. Hans Beckhoff

Registergericht:  
 Gütersloh HRA 7075

Ust.-Id.-Nr.: DE 815529334

Kreissparkasse Wiedenbrück  
 BIC: WELADED1WDB  
 IBAN: DE24 4785 3520 0000 0600 04 (EUR)  
 IBAN: DE02 4785 3520 0000 0600 12 (USD)  
 Deutsche Bank (EUR, USD)  
 BIC: DEUTDE33HAN  
 IBAN: DE95 4807 0043 0373 6014 00