

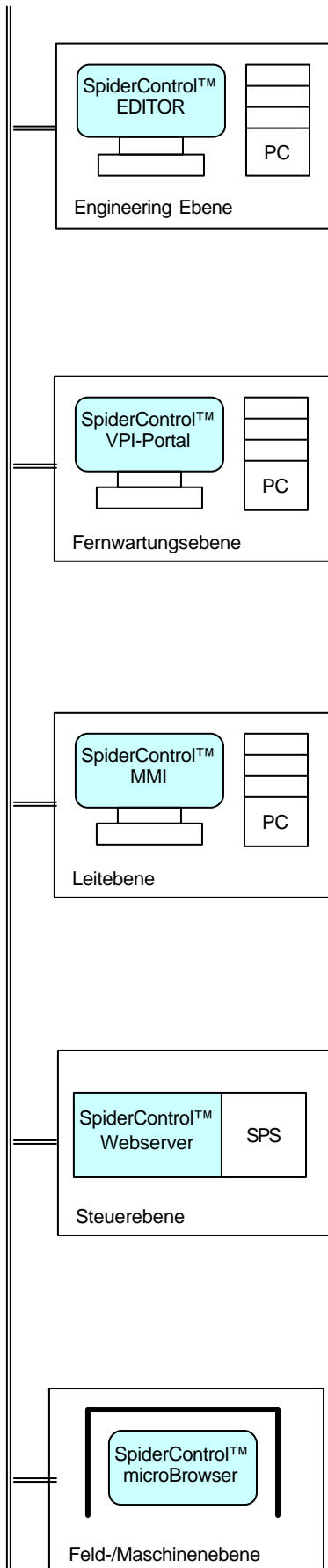


SpiderControl™ EDITOR

PLC Edition
Version 4.0



iniNet AG. Erfolgreiche Kunden
E-mail: info@ininet.ch, www.ininet.ch, Tel: +41 61-716 96 26



TCP/IP

Übersicht

Vielen Dank dass Sie SpiderControl™ EDITOR gewählt haben. SpiderControl™ ist die fortschrittlichste Technologie für webbasierte grafische Benutzerschnittstellen (MMIs), mit einem in der Praxis ausgereiftem Entwicklungswerkzeug. SpiderControl™ basiert auf offenen JAVA-Applet Standards.

Webbasierte grafische Benutzerschnittstellen (MMIs) sind von überall her erreichbar. Sei es Vorort mittels Touchpanels, in der Leitwarte via PC und Internet Explorer oder von Fern mittels Telefonwählverbindungen , WLAN, Internet Router, etc..

Mehrere MMI von dezentralisierten Systemen, können zentral in einem SpiderControl™ VPI-Portal zusammengefasst werden.

SpiderControl™ ist praktisch für alle gängigen SPSen verfügbar sowie ausserdem für verschiedenste Microcontroller-Plattformen im „embedded Systems“ – Bereich. Nehmen Sie mit uns Kontakt auf für spezifische Informationen über Ihre Zielplattform.

Der SpiderControl™ EDITOR erlaubt eine rein grafische Entwicklung von SpiderControl™ MMIs. Die Kommunikation zwischen MMI und Webserver/Zielsystem ist bereits programmiert.

Ein normaler WEB-Browser mit JAVA VM 1.1 Unterstützung wie z.Bsp.: MS Internet Explorer, Mozilla, Opera,... genügt zur Visualisierung. Dadurch kann auf eine Vielfalt von Panels oder PCs zugegriffen werden.

Der Browser kommuniziert mit dem Webserver via TCP/IP – Protokoll. Dieses kann über verschiedene Medien wie Ethernet, Modem, Wireless LAN, etc gerouted werden

Durch die zentrale Datenhaltung der MMIs als JAVA-Applet auf der SPS gibt es keine Versionskonflikte bzw. Code-Redundanzen zwischen Steuerungssoftware und Visualisierungssoftware. Dadurch reduziert sich der Entwicklungs- und Unterhaltsaufwand erheblich und es resultiert ein echtes objektorientiertes Design.

Das Visualisierungssystem kann ausgetauscht werden ohne dass eine Neuprogrammierung notwendig wird. Auch die SPS kann ausgetauscht werden ohne Anpassungen im MMI.

Für den Bereich der Vorort-Bedienpanels (Touchpanels, Operating Panels) ist als OEM-Produkt ein SpiderControl™ microBrowser verfügbar, mit dessen Hilfe sich einfachste und dadurch kostengünstige Microcontroller-basierte Displaylösungen aufbauen lassen. Kontaktieren Sie uns für weitere Informationen.

Sämtliche Software wurde von iniNet Solutions GmbH entwickelt und kann daher auf praktisch jedes beliebige Zielsystem portiert werden.



SpiderControl™ EDITOR PLC Edition

Für die Projektierung von MMIs mithilfe von SpiderControl™ EDITOR PLC Edition sind keinerlei Programmierkenntnisse erforderlich. Der Ablauf der Projektierung geschieht in drei Schritten:

1. Zeichnen der Grafikbilder mithilfe von Grafiksymbolen. Diese können mit Prozessdatenpunkten (PDPs) der SPS-Software dynamisch verbunden werden (Anzeige und Änderung von PDPs, Farbumschlag bei Alarmmeldungen). Die Verbindung basiert auf symbolischen Namen welche nur innerhalb der Grafikbilder sichtbar sind und im nächsten Schritt mit den wirklichen SPS-Daten verbunden werden. Aus den Grafikbilder wird dann die HTML-Startseite erzeugt, die sich bereits mit einem PC-Browser kontrollieren lässt.
2. Nun müssen die MMI internen symbolischen Namen mithilfe des Dialoges *InitPPO* mit den wirklichen Prozessdatenpunkten (Ein-, Ausgänge, Merker, Variablen, ..) verbunden werden.
3. Als letztes müssen sämtliche benötigten Dateien in die SPS übertragen werden. Dies ist SPS-abhängig und wird im Kapitel *Initialisierung der Prozessdatenpunkte* beschrieben.

Dateiformate

Dieser Abschnitt erklärt alle Dateien welche zur Erzeugung von SpiderControl™ basierten MMIs benötigt werden.

Der Browser adressiert eine HTML-Seite mit

http://<ziel_system_ip_adresse>/<ihr_projekt>.html.

Diese Datei wird durch den SpiderControl™ EDITOR erzeugt (Menu: Projekt->Generate HTML..). Sie enthält eine Referenz auf ein Java-Applet welches die Visualisierung übernimmt. Weitere Referenzen zeigen auf *.teq – Dateien, die vom SpiderControl™ EDITOR für jedes Grafikbild erzeugt werden und deren Layout bestimmen. Sämtliche JAVA Klassen werden in einem JAVA-Archiv mit dem Namen *IMaster.jar* gespeichert.

Um nun eine embedded Webserver basierte Visualisierung zu erhalten, müssen folgende Dateien im Zielsystem gespeichert werden:

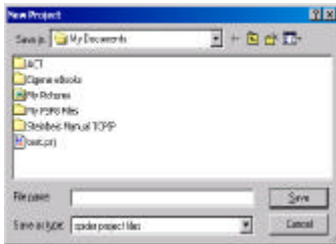
- Die <ihr_projektnamen>.html -Startseite
- Die JAVA Klassenbibliothek *IMaster.jar*
- Alle Grafikbilder (*.teq – Dateien)
- Alle zusätzlichen Anwenderspezifischen HTML-Dateien (wie z.Bsp: Hilfeseiten, ..)
- Alle Bilddateien die benutzt werden.
- Die Variablen-Initialisierungsdateien <ihr_projektnamen>.tcr und <ihr_projektnamen>.itq

Es ist darauf zu achten, dass bei allen selbstgeschriebenen Webseiten für Referenzen keine Pfadangaben verwendet werden, da keine Unterverzeichnisse unterstützt werden.

Erste Schritte

Nach dem Aufstarten von SpiderControl™ EDITOR erscheint das Hauptfenster, das in zwei Bereiche unterteilt ist. Im linken Bereich erscheinen alle geöffneten Dateien. Im rechten Bereich werden die Grafikbilder angezeigt und lassen sich dort editieren.

Erstellen eines Projektes



Wählen Sie den Menübefehl `Datei->Neues Projekt` aus und Sie erhalten nebenstehenden Dialog. Legen Sie ein neues Verzeichnis an, gehen Sie in dieses Verzeichnis und geben Sie den gewünschten Projektnamen an. Drücken Sie „Speichern“

Eine Projektdatei `*.prj` sowie sämtliche sonst benötigten Dateien werden erzeugt. Alle diese Dateien, sowie jene die später dem Projekt hinzugefügt werden, werden im linken Teil des Hauptfensters aufgelistet.

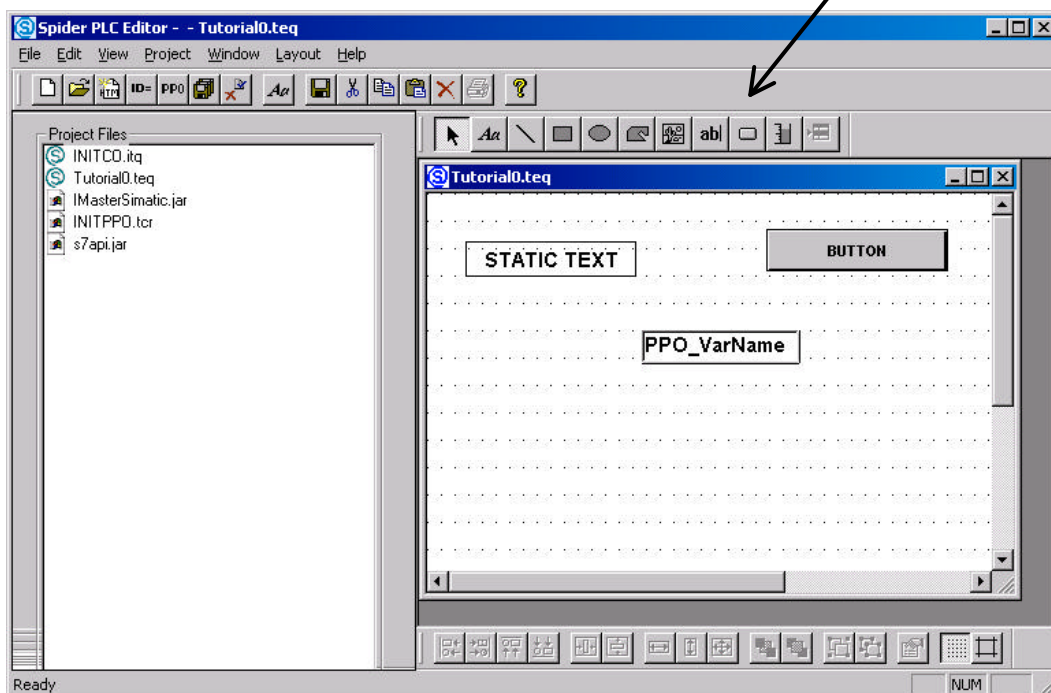
Erstellen eines Grafikbildes

Im nächste Schritt werden wir ein leeres Grafikbild dem Projekt hinzufügen. Wählen Sie den Menübefehl `Projekt->Hinzufügen->Neu...`. Der Editor schlägt einen Dateinamen vor, der aus dem Projektnamen, einem fortlaufendem Index und der Endung `*.teq` besteht

`<ihr_projektname0>.teq`

Nun können Sie anfangen grafische Visualisierungsobjekte einzufügen. Diese wählen Sie aus der Symbolleiste für Grafiksymbole aus. Die Bedeutung dieser Symbole wird im Kapitel „*Grafiksymbol Referenz*“ beschrieben.

Symbolleiste für Grafiksymbole



Hinzufügen eines Textes – Anzeigen eines PDP's



Wählen Sie aus der Symbolleiste für Grafiksymbole ein Textfeld aus und platzieren ihn im Grafikbild. Dies geschieht durch Setzen der linken oberen Ecke mittels linker Maustaste und durch Setzen - bei gedrückter Maustaste - der rechten unteren Ecke. Geben Sie die Maustaste frei und das Symbol ist platziert.

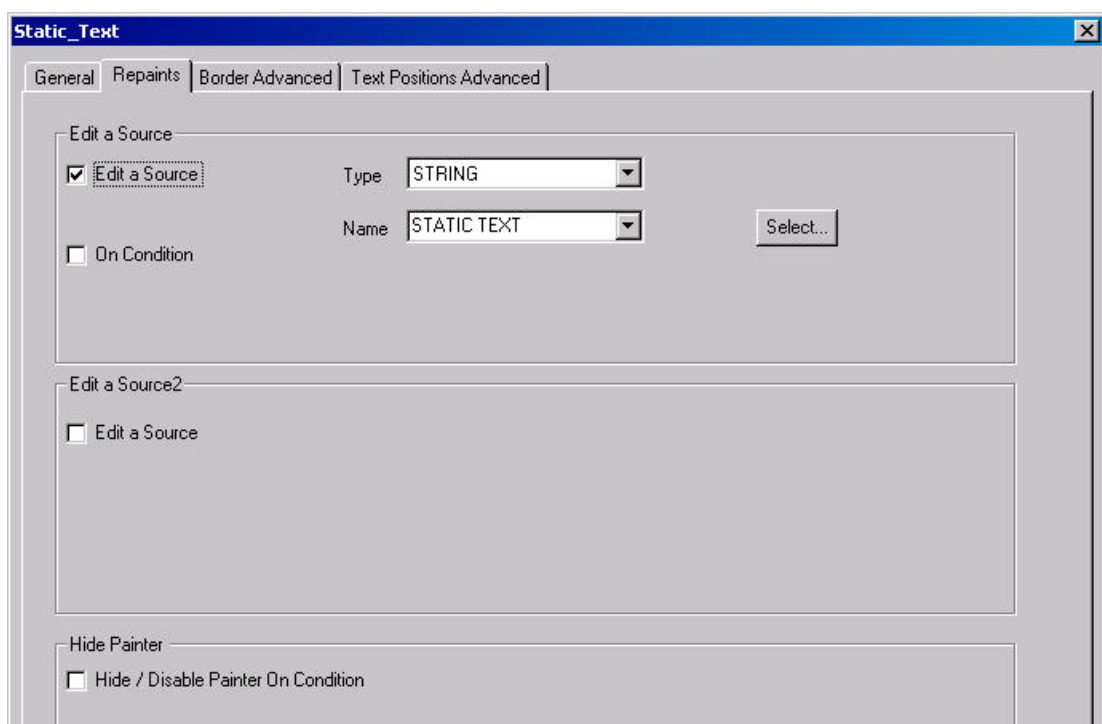
Durch Doppelklicken auf das Symbol erscheint der Eigenschaften-Dialog. Er enthält verschiedene Seiten mit Eigenschaften des Symbols. Mit der ersten Seite Allgemein werden statische Eigenschaften wie Position, Grösse, Farben und Schriftart festgelegt. Diese Eigenschaften existieren für alle Grafiksymbole.

Mit der zweiten Seite Darstellung lässt sich einstellen, was das Symbol anzeigen soll. Aktivieren Sie das Kontrollkästchen Datenquelle, wählen Sie STRING im Auswahlfeld Typ und geben Sie den gewünschten Text im Feld Name ein. Drücken Sie OK und der Text erscheint im Grafikbild.

Alternativ lässt sich der Typ eines Grafiksymbols auf HTML_TAG, CONTAINER oder PPO setzen. Mit PPO (=PDP Prozessdatenpunkt) als Typ lassen sich auf diese Weise Prozessdatenpunkte darstellen. Wenn Sie einen PDP nicht nur ausgeben aber auch editieren wollen, wählen Sie das Grafiksymbol Eingabefeld, dass im nächsten Abschnitt beschrieben wird.

Mit dem Typ CONTAINER lässt sich eine lokale Variable darstellen. Lokale Variablen sind nur innerhalb eines oder mehrerer Grafikbilder sichtbar.

Mit dem Typ HTML_TAG kann ein Link zu einer beliebigen HTML-Seite angezeigt werden.



Ändern eines Prozessdatenpunktes (PDP)

abl

Im nächsten Schritt fügen wir dem Grafikbild ein Eingabefeld hinzu. Dieses Objekt erlaubt die Darstellung und Eingabe eines Prozessdatenpunktes. Wählen Sie das Eingabefeld aus der Symbolleiste für Grafiksymbbole und platzieren Sie es auf dieselbe Weise wie ein Textfeld-Symbol. Öffnen sie das Eigenschaftsfenster durch einen Doppelklick auf das Symbol und gehen Sie auf die Seite Darstellung.

Das Datenquelle Kontrollkästchen muss aktiviert sein und der Typ sollte auf PPO gesetzt sein. Im Namensfeld können Sie nun einen symbolischen Namen für Ihren Prozessdatenpunkt angeben. Dieser Name ist nur innerhalb eines SpiderControl™ MMI Projektes sichtbar und muss später mit dem richtigen Prozessdatenpunkt auf der Steuerung verknüpft werden. Dies geschieht durch die Menufunktion Projekt->InitPPO..

Erzeugung eines zweiten Grafikbildes

Ein zweites Grafikbild soll nun auf die gleiche Art hinzugefügt werden. Wählen Sie wiederum den Menubefehl Projekt->Hinzufügen->Neu... Übernehmen Sie den vorgeschlagenen Dateinamen oder wählen Sie einen eigenen. Fügen Sie nun dem neuen Grafikbild einen statischen Text der Art „Dies ist das zweite Bild“ hinzu.

Navigation zwischen den Grafikbildern



Im nächsten Schritt wollen wir Bedienknöpfe einzeichnen, mit denen wir zwischen zwei Grafikbildern navigieren können. Wählen Sie aus der Symbolleiste für Grafiksymbbole das Symbol für einen Bedienknopf aus und zeichnen Sie einen in das erste Grafikbild. Ein Doppelklick auf das gezeichnete Symbol öffnet auch hier den Eigenschaftsdialog. Geben Sie dem Bedienknopf einen Namen und formatieren ihn mit den Eigenschaften der Dialogseite Darstellung auf die gleiche Art und Weise wie ein Textfeld-Symbol. Anschließend gehen Sie auf die Dialogseite Aktionen und aktivieren das Kontrollkästchen Bildwechsel. Geben Sie im Feld Bildname den Namen Ihres zweiten Grafikbildes ein.



Auf gleiche Weise können Sie nun im zweiten Grafikbild einen Bedienknopf einfügen und dort die Aktion Bildwechsel auf ihre

erste Seite setzen. Nun können Sie zwischen den beiden Seiten navigieren. Alternativ können Sie mit `Letztes Bild` auf die jeweils letzte gewählte Seite wechseln oder einen `Hyperlink` konfigurieren und damit auf eine beliebige Webseite springen.

Erzeugen der HTML-Startseite

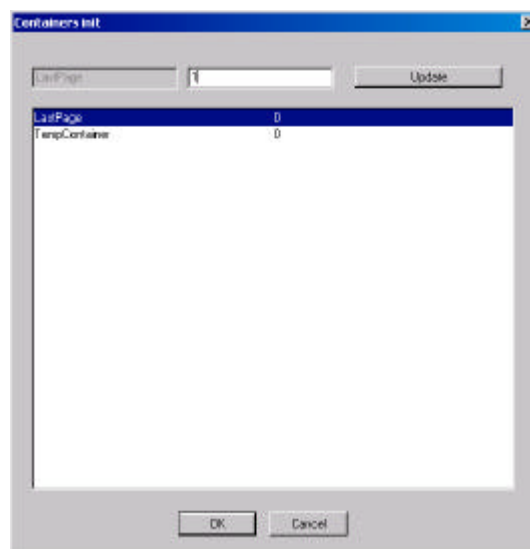
Im letzten Schritt generieren wir eine HTML-Datei, die das ganze Projekt umfasst und als Startseite für unser MMI dient. Wählen Sie den Menübefehl `Projekt->Erzeuge HTML...`. Übernehmen Sie den vorgeschlagenen Dateinamen oder wählen Sie einen eigenen. Anschliessen sollten Sie alles mit dem Menübefehl `Datei->Alles speichern` speichern.

Erster Test in einem Browser

Für einen ersten Test lässt sich die Datei `Tutorial.html` mit einem PC-Browser (MS Internet Explorer, Netscape Navigator, Mozilla, Opera, ...) öffnen und testen. Natürlich werden hier aber noch keine Prozessdatenpunkte angezeigt. Diese müssen erst noch mit den symbolischen Namen des Projektes verknüpft werden. Anschliessend muss das Projekt in den Webserver im Zielsystem übertragen werden. Dies ist Zielsystemabhängig und wird im übernächsten Kapitel beschrieben. Des weiteren müssen noch die MMI internen Variablen initialisiert werden. Diese heissen Container und sind nur im MMI sichtbar und nicht im Zielsystem.

Initialisierung der Container

Container sind lokale Variablen, die nur innerhalb eines oder mehrerer Grafikbilder sichtbar sind. Mit dem Menübefehl `Projekt->Init Container...` lässt sich der Eigenschaftsdialog der Container öffnen. Darin werden alle Container aufgelistet. Durch Selektion eines Containers wird sein Wert in das obere Eingabefeld übertragen und lässt sich dort ändern. Mit dem Bedientknopf `Aktualisieren` wird der Wert übernommen. Ein Klick auf den Bedientknopf `Ok` schliesst den Dialog und erzeugt eine Container Initialisierungsdatei mit dem Namen `<ihr_projektname>.itq`. Diese Datei muss auch ins Zielsystem übertragen werden.



Initialisierung der Prozessdatenpunkte

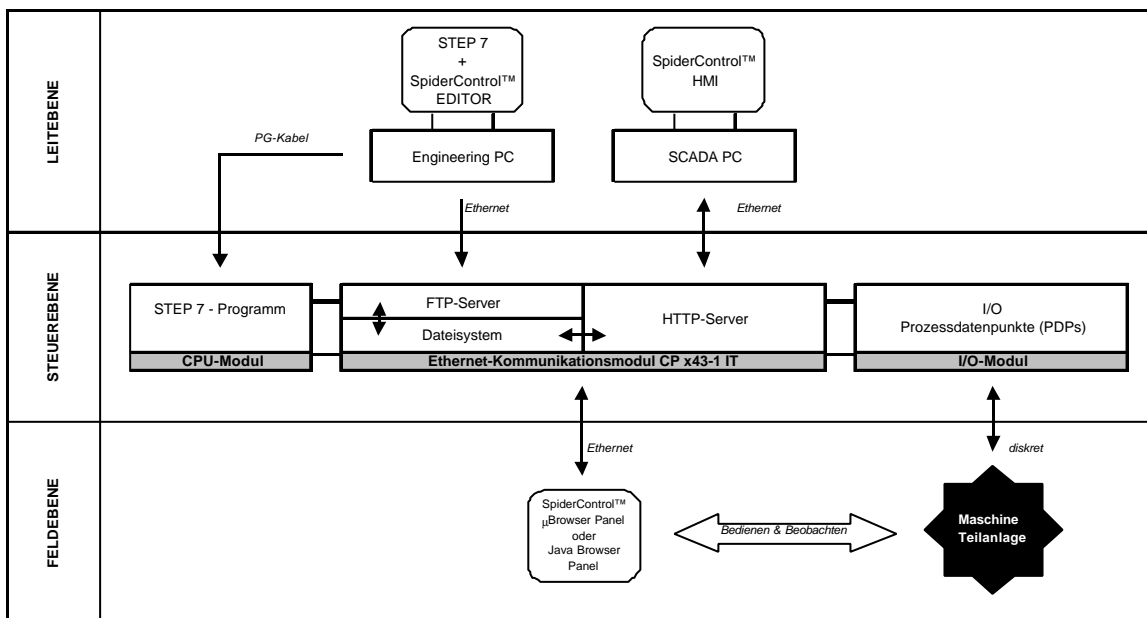
SpiderControl™ SIMATIC Konzept

SpiderControl™ wurde für die SPSen der Reihe SIMATIC (Siemens) speziell angepasst, um die vorhandenen Möglichkeiten der SIMATIC zu nutzen. SpiderControl™ SIMATIC ist kompatibel zu den Baureihen S7/200, S7/300 und S7/400. Als Ethernetschnittstelle, Webserver (HTTP-Server) sowie Dateisystem dient das SIMATIC Kommunikationsmodul CP x43-1 IT (x=2,3,4).

Als Entwicklungswerkzeug dient SpiderControl™ EDITOR PLC Edition 4 SIMATIC. Die damit erzeugten MMI-Dateien werden via FTP-Server des Kommunikationsmoduls auf dessen Dateisystem gespeichert. Dies können HTML-Seiten, Textdateien, Java Klassen, GIF-Bilder sowie SpiderControl™ Grafikbild-Dateien und Variablen-Zuweisungsdateien sein.

Das SpiderControl™ JAVA-Applet, das die ganze Visualisierung und Bedienung des MMI übernimmt, nutzt die vorhandenen Ressourcen des Kommunikationsmoduls mittels Siemens JAVA-Beans, die auf dem CP x43-1 IT vorhanden sind.

Dadurch kann aus dem MMI direkt auf die Prozessdatenpunkte (PDP) wie Eingänge, Ausgänge, Merker und Datenbausteine zugegriffen werden. Verschiedenste Datenformate wie BIT, BYTE, WORD, REAL, STRING stehen dabei zur Verfügung.



Dabei werden innerhalb der SpiderControl™ Grafikbilder die Prozessdatenpunkte nicht direkt und absolut adressiert, sondern es werden SpiderControl™ MMI interne symbolische Namen verwendet.

Die Zuordnung dieser symbolischen Namen zu einem realen SIMATIC Prozessdatenpunkt erfolgt mit einem in den SpiderControl™ EDITOR integrierten Tool (InitPPO). Dieses Tool erzeugt eine ASCII-Datei <ihr_projektname>.tcr, dass die Verbindung zwischen STEP7-Programm und SpiderControl™ MMI definiert.

InitPPO SIMATIC Dialog

Nach Erstellung der Grafikbilder müssen noch sämtliche Referenzen, die in den Grafiksymbolen als symbolische Namen verwendet werden, mit den Prozessdatenpunkten verknüpft werden. Dazu steht der Menübefehl **Projekt->InitPPO** zur Verfügung welcher folgenden Dialog öffnet.

Symbolischer Name	PPO Min Val	PPO Max Val	PPO format Val	PPO unit Val	PPO Read type	S7 Var Type
RaumTemp	2	2	1	129	0 0	0
Befehl_Aus	2	1	1	130	0 0	1
Befehl_Ein	2	1	1	130	0 0	0
Betrieblunden	2	2	1	129	0 0	0
Energieverbrauch	2	2	1	129	0 0	0
SollTemp	2	2	1	129	0 0	0
Status	2	2	1	129	0 0	0
Stromaufnahme	2	2	1	129	0 0	0
WTRH1	2	2	1	129	0 0	0

Hier werden sämtliche verwendete symbolische Namen mit ihrer Parametrierung aufgelistet. Nun lässt sich jeder Name anwählen, worauf seine Parameter in den oberen Teil übertragen werden, wo sie geändert und mit dem Bedientaste Aktualisieren übernommen werden können. Die Parameter lauten wie folgt:

PPO min Val & PPO max Val

Für Grafiksymbole des Typs **Eingabefeld**, lassen sich hier die untere und obere Grenze angeben. Bei Unter- oder Überschreitung meldet SpiderControl™ eine Fehleingabe.

PPO format Val

Kann sein **int** oder **string**. Bestimmt die Formatierung der Bildschirmausgabe eines Prozessdatenpunktes.

PPO Unit Val

Die Einheit eines Prozessdatenpunktes.

PPO Read Type

Kann sein `individual` oder `MB_Block`. Mit dem ersteren wird dieser Prozessdatenpunkt einzeln vom Zielsystem angefragt. Mit `MB_Block` werden alle PDPs gemeinsam angefragt. Dies spart Kommunikationszeit.

S7 Var Type

Hier lassen sich verschiedene S7 Datentypen angeben, so wie sie in der Dokumentation zur Siemens Kommunikationseinheit beschrieben werden (Stichwort: Java-Beans, S7Put/GetApplets).

S7 Rep. Faktor

Hier lässt sich angeben wieviele Variablen gelesen werden sollen. Bei S7 Strings muss hier die Stringlänge angegeben werden.

S7 Var Mem Area

Hier parametriert man den Zugang auf Merker, Datenbausteine sowie Ein- oder Ausgänge.

S7 Var Sub Area

Bei Datenbausteinen muss hier die Nummer eingegeben werden. Für S7/200 gibt es nur DB1.

S7 Var Byte Offset

Die Merkenummer, DB-Variablenoffset, Ein-/Ausgangsgruppe.

S7 Var Bit Offset

Für den Zugriff auf einzelne Bits.

Datenübertragung in die SIMATIC S7

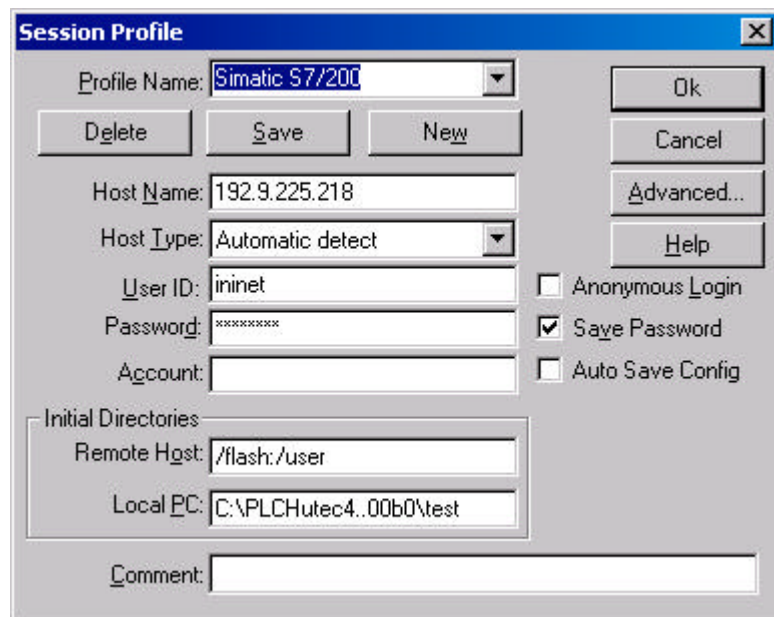
Sämtliche mit SpiderControl™ EDITOR erzeugten oder referenzierte Ressourcen wie HTML-Seiten, JPEG-Bilder, *.teq-Dateien, IMasterSimatic.jar usw. befinden sich im Projektverzeichnis und müssen von dort in die Kommunikationseinheit CPx34 heruntergeladen werden.

Dazu wird der FTP-Server der SIMATIC Kommunikationseinheit benutzt. Dieser muss zuerst mithilfe der SIMATIC Step 7 Programmierumgebung konfiguriert werden. Dafür steht im Menü „Extras“ ein „Internet-Assistent“ zur Verfügung.

Mit diesem „Internet-Assistent“ lassen sich der Kommunikationseinheit eine IP-Adresse mit Subnet-Maske zuweisen. Des Weiteren muss ein Administrator-Konto eingerichtet werden um sich damit auf dem FTP-Server einloggen zu können.

Für die eigentliche Datenübertragung können Sie ein beliebiges FTP-Tool für Windows benutzen, wie zum Beispiel WS_FTP, das dem SpiderControl™ EDITOR Paket beiliegt.

Nach dem Starten von WS_FTP erscheint folgender Dialog:



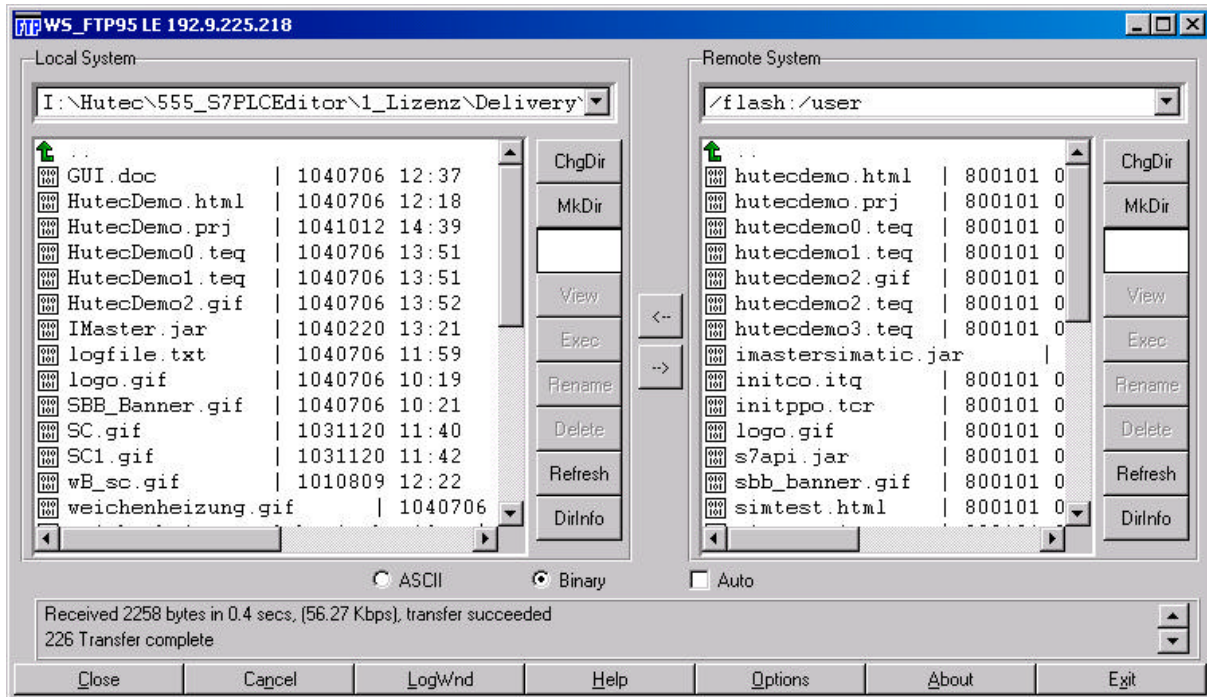
Unter „Host Name“ geben Sie die IP-Adresse Ihrer Kommunikationseinheit an. „User ID“ und „Password“ entsprechen dem Administrator-Konto das Sie für die Kommunikationseinheit eingerichtet haben. Als „Initial Directories“ geben Sie für „Remote Host“ den Pfad auf dem FTP-Server Ihrer Kommunikationseinheit an. Verwenden Sie hier „/flash/user“. Unter „Local PC“ geben Sie den Pfad Ihres SpiderControl™ EDITORs Projektverzeichnisses an. Mit „Save“ können Sie die gemachten Einstellungen unter einem eigenen „Profile Name“ abspeichern und mit „Ok“ den Dialog beenden.

Im nun aktiven Hauptfenster von WS_FTP erscheint links das Quellverzeichnis Ihres SpiderControl™ EDITOR Projektes und rechts der Inhalt des FTP-Servers. Mit dem Bedienknopf „->“ in der Mitte, lässt sich nun die Dateiübertragung starten.

Nach erfolgter Dateiübertragung lässt sich Ihr Projekt bereits starten. Geben Sie dazu in einem Internet-Browser (IE Explorer o.ä.) folgendes ein:

http://<ip-adresse-kommunikationseinheit>/user/<ihr_projektname>.html.

Beispiel: <http://192.9.225.218/user/hutecdemo.html>.



Aktualisierung der Prozessdatenpunkte

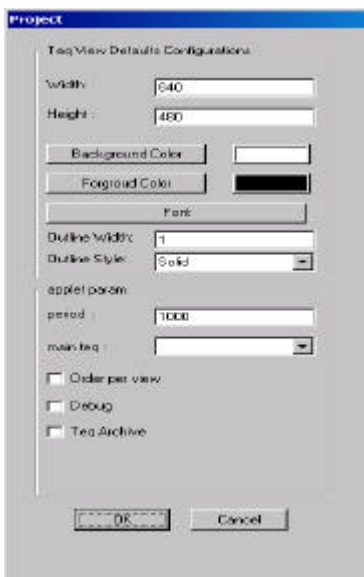
Aktualisierung sämtlicher PDPs auf einmal

Jedes Grafikbild kann mehrere Prozessdatenpunkte (PDPs) enthalten. Um eine effiziente Aktualisierung zu erhalten, werden standardmässig sämtliche PDPs aller Grafikbilder auf einmal beim Zielsystem (SPS) nachgefragt. Dies vermindert den Kommunikationsaufwand und führt zu einer schnelleren Initialisierung eines neuen Grafikbildes, da alle benötigten Werte bereits bekannt sind.

Aktualisierung der PDPs pro Grafikbild

Alternativ lässt sich die Option `Aktualisierung pro Bild` aktivieren. Diese findet sich im Dialog `Projekt->Projekt Einstellungen...` Damit werden nur diejenigen Prozessdatenpunkte beim Server nachgefragt werden, die für das aktive Grafikbild benötigt werden. Dies kann bei einer grossen Anzahl von zu aktualisierenden PDPs die effizientere Methode sein.

Aktualisierungsperiode



Standardmässig werden Prozessdatenpunkte periodisch beim Server nachgefragt und aktualisiert. Im Dialog `Projekt->Projekt Configurations` lässt sich diese Periode einstellen.

Die SpiderControl™ Technologie erlaubt auch eine ereignisgesteuerte Aktualisierung der PDPs. Dies ist ein zusätzliches Feature. Kontaktieren Sie uns bei Bedarf.

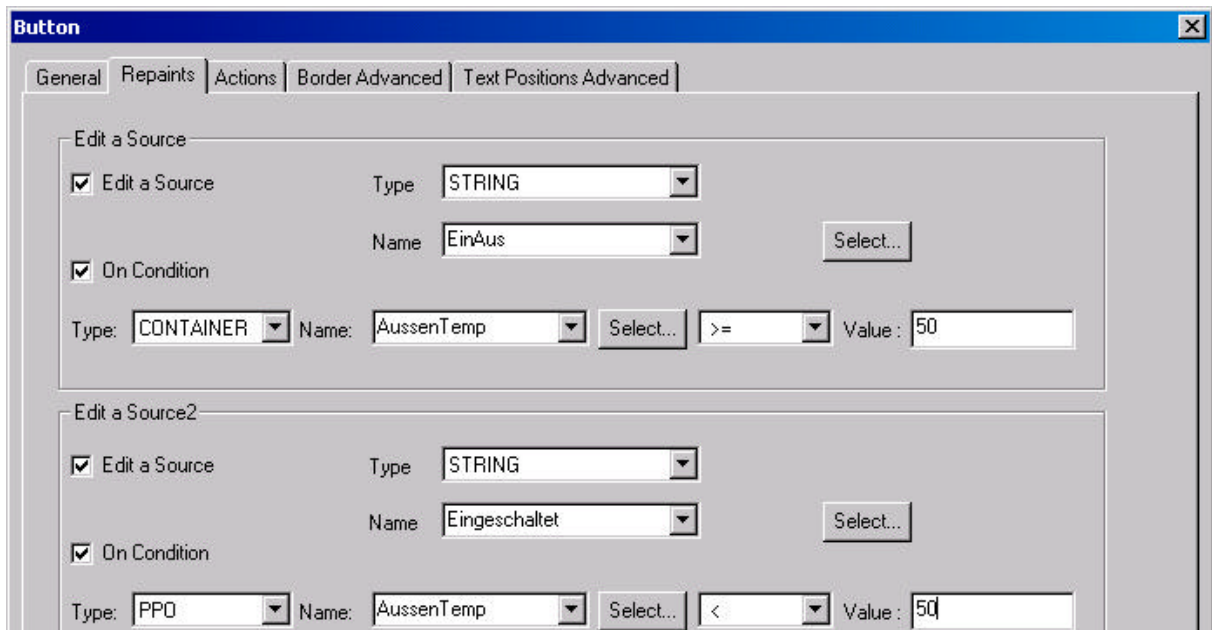
Aktionen in Abhängigkeit von Variablen

Gewisse Eigenschaften der Grafiksymbbole lassen sich mit einer Bedingung verknüpfen wie zum Beispiel die Datenquellen in der Eigenschaftsseite Darstellung oder der Farbumschlag auf der Eigenschaftsseite Farbumschlag. Das Format dieser Bedingungen ist immer gleich und wird hier beschrieben.

Zuerst wird im Feld `Typ` die Art der Variablen ausgewählt von der die Bedingung abhängen soll. Diese kann vom Typ `PPO` (Prozessdatenpunkt) oder vom Typ `CONTAINER` (MMI interne Variable) sein. Anschliessend wird der Name der Variablen im Feld `Name` angegeben. Zuletzt muss der Wert mit dem die Variable verglichen wird im Feld `Wert` eingegeben werden und die Art des Vergleiches. Dafür stehen folgende Möglichkeiten zur Auswahl:

- != ungleich
- < kleiner als
- <= kleiner als oder gleich
- == gleich

- >= grösser als oder gleich
- > grösser als



MMI mit mehreren Grafikbildern

Dieser Abschnitt erklärt die beiden Möglichkeiten ein MMI mit mehreren Grafikbildern zu generieren. Entweder man packt alle Bilder in ein JAVA Applet oder man arbeitet mit HTML-Links zu den verschiedenen Bildern.

Mehrere Grafikbilder in einem JAVA Applet

Im Normalfall greift der Benutzer via Browser auf eine einzige HTML-Seite zu. Diese wird durch die Menubefehl *Projekt->Erzeuge HTML..* erzeugt und hat typischerweise den Namen

```
<ihr_projektname>.html
```

Diese Seite referenziert nun ein JAVA-Applet, dass die SpiderControl™ Technologie enthält und den Namen

```
IMaster<zielsystem>.jar
```

besitzt. Dieses Applet referenziert nun sämtliche Grafikbilder (* .teq Dateien), die zum MMI gehören. Dies ist der schnellste Weg, da alle Bilder sowie das Applet beim ersten Zugriff geladen werden.

Der Ausdruck <zielsystem> ist von dem Zielsystem (SPS) abhängig. Folgende Abbildung zeigt den HTML-Code der Startseite, die mit *Projekt->Erzeuge HTML..* erzeugt wurde.

```

<!-- HTML FILE GENERATED FROM PLC EDITOR by ININET Solutions Gmbh on Wednesday, July 14, 2004 -->
<HTML>
<HEAD>
  <META HTTP-EQUIV="Content-Type" CONTENT="text/html;CHARSET=iso-8859-1">
  <META NAME="GENERATOR" CONTENT="IniNet SpiderControl TM">
  <TITLE>Tutorial</TITLE>
</HEAD>
<BODY bgcolor="#FFFFFF">
  <p>
<APPLET ARCHIVE="IMaster.jar" CODE="IMaster.class" ALIGN="BOTTOM" WIDTH="640" HEIGHT="480">
  <PARAM NAME="AppletName" VALUE="Tutorial">
  <PARAM NAME="ProjectName" VALUE="Tutorial">
  <!--PARAM NAME="DrawTeqsAtInit" VALUE="true"-->
  <PARAM NAME="Period" VALUE="1000">
  <PARAM NAME="FirstFileIndex" VALUE="0">
  <PARAM NAME="LastFileIndex" VALUE="2">
  <PARAM NAME="Main" VALUE="0">
  <PARAM NAME="OrderPerView" VALUE="off">
  <PARAM NAME="debug" VALUE="off">
  <PARAM NAME="SpiderHTMLVersion" VALUE="2">
  <PARAM NAME="MainTEQName" VALUE="Tutorial0">
  <PARAM NAME="TEQ0NAME" VALUE="Tutorial0">
  <PARAM NAME="TEQ1NAME" VALUE="Tutorial1">
  <PARAM NAME="TEQ2NAME" VALUE="Tutorial2">
  <PARAM NAME="InitCoTEQName" VALUE="INITCO.itq">
  <PARAM NAME="InitPPOASCIName" VALUE="INITPPO.tcr">
</APPLET>
  </p>
</BODY>
</HTML>

```

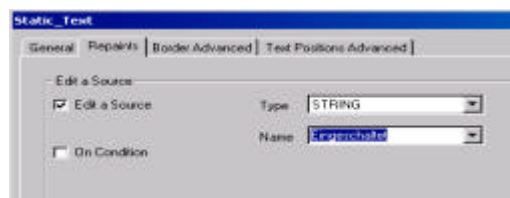
Mehrere Grafikbilder mit HTML-Links

Es ist auch möglich aus der Startseite `<ihr_projektname>.html` mit HTML-Links auf weitere zu verweisen. Der Nachteil ist aber, dass dann diese Seiten erst bei Aufruf geladen werden, was zu einer Verzögerung in der Darstellung führen kann. Für folgende Fälle kann dies aber durchaus nützlich sein:

- Links zu Hilfedateien (als HTML gespeichert)
- Links zu MMI's von anderen Webservern, um ein gemeinsames MMI zu bilden
- Links zu anderen Applets oder Seiten, die nicht mit dem SpiderControl™ EDITOR erzeugt wurden.

MMI mit mehreren Sprachen

Um mehrsprachige MMIs zu unterstützen, gibt es als Datenquelle den Typ `HTML-TAG` als Ersatz zum Typ `STRING` den Sie im Eigenschaftsdialog auf der Seite Darstellung verwenden können.



Falls Sie wie in der oberen Abbildung die Datenquelle mit Typ = `STRING` und Name = „Eingeschaltet“ konfigurieren, so erhalten Sie in Ihrer MMI immer den Text „Eingeschaltet“. Falls Sie aber wie unten abgebildet die Datenquelle mit Typ = `HTML-TAG` und Name = „Eingeschaltet“ konfigurieren, so erhalten Sie, wenn Sie nichts anderes machen, auch den Text „Eingeschaltet“ in Ihrem MMI.

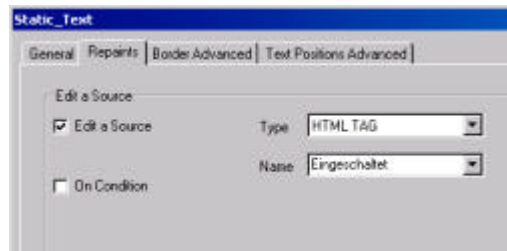
Sie können nun aber in der HTML-Startseite

```
<ihr_projektname>.html
```

manuell Zeilen der Form

```
<PARAM NAME="<Eingeschaltet>" VALUE="On">
```

einfügen. Je nach Value wird nun der Text „On“ in Ihrem MMI dargestellt. Mit verschiedene Versionen der Startseite für jede Sprache lassen sich so mehrsprachige MMIs realisieren.



Grafiksymbol Referenz

Sämtliche verfügbaren Grafiksymbole sind in der Grafiksymbol Symbolleiste enthalten



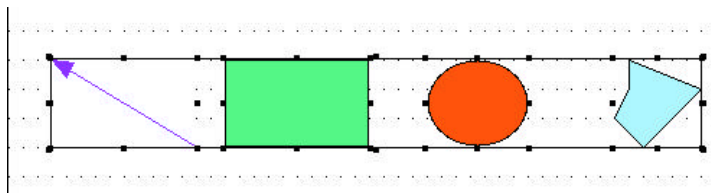
Die Grafiksymbole haben unterschiedliche Eigenschaften, die über den Eigenschaftsdialog konfiguriert werden können. Dieser lässt sich mithilfe eines Doppelklicks auf das Symbol öffnen. Die einzelnen Eigenschaften sind im Kapitel „Eigenschaften-Dialog“ beschrieben.

Statische Grafiksymbole

Rechteck, Ellipse, Polygon



Mit diesen Grafiksymbolen lassen sich keine Prozessdatenpunkte oder Container anzeigen. Es sind reine Zeichnungselemente, deren Farbeigenschaften sich aber mit einem PDP oder Container verknüpfen lassen, um einen Farbumschlag zu realisieren. Dies lässt sich im Eigenschaftsdialog auf der Seite *Darstellung* und *Farbumschlag* definieren.



Line



Das Liniensymbol besitzt die gleichen Eigenschaften wie Rechteck, Ellipse und Polygon. Zusätzlich lässt sich über die Eigenschaftsseite *Linie* angeben, ob der Linienanfang oder das Ende mit einem Pfeil gezeichnet werden soll.

Image



Mit diesem Grafiksymbol lässt sich ein Bild anzeigen, das als externe Datei mit dem Symbol verknüpft wird. Als Dateiformat wird GIF unterstützt. In der Eigenschaftsseite *Darstellung* können sogar zwei verschiedene Bilder in Abhängigkeit einer Bedingung angezeigt werden. Die Dateien müssen ins Zielsystem kopiert werden. Das Bild wird auf die Grösse des Grafiksymbols skaliert und falls das Bild einen transparenten Hintergrund hat, wird die Hintergrundfarbe des Grafiksymbols sichtbar.



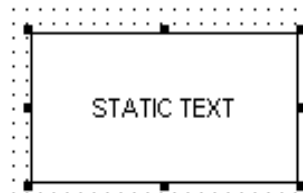
Dynamische Grafiksymbole

Dynamische Grafiksymbole lassen sich mit einem Prozessdatenpunkt (PDP) oder mit einer Container-Variablen verknüpfen. Dadurch lassen sich PDPs/Container anzeigen und verändern. Wie die statischen Grafiksymbole lässt sich auch hier die Farbdarstellung in Abhängigkeit eines PDPs/Containers konfigurieren.

Textfeld



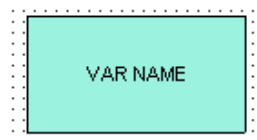
Ein Textfeld dient zur Ausgabe eines statischen Textes, eines Prozessdatenpunktes oder einer Container-Variablen. Dies lässt sich in der Eigenschaftsseite *Darstellung* im Feld *Typ* einstellen. Die Farbdarstellung sowie die Textausrichtung lassen sich auch mit dem Eigenschaftsdialog konfigurieren.



Eingabefeld



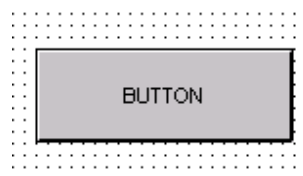
Mit einem Eingabefeld lässt sich ein Prozessdatenpunkt (PDP) oder eine Container-Variablen verändern. Dies lässt sich in der Eigenschaftsseite *Darstellung* im Feld *Typ* einstellen. Als Name für den PDP/Container wird ein symbolischer Name verwendet, der mit dem Dialog *Init PPO* mit dem wahren PDP des Zielsystems verknüpft wird.



Bedienknopf



Mit einem Bedienknopf lassen sich Prozessdatenpunkte setzen, Grafikbilder wechseln, externe Webseiten aufrufen oder Variablen inkrementieren. All dies lässt sich mit der Eigenschaftsseite *Aktionen* konfigurieren. Für die Darstellung des Bedienknopfes lassen sich sämtliche Typen wie *CONTAINER*, *PPO*, *FILE*, *STRING* und *HTML-TAG* verwenden.



Fortschrittsbalken



Mit einem Fortschrittsbalken lässt sich ein Prozessdatenpunkt oder eine Container-Variable grafisch und dynamisch darstellen. Der Balken hat einen Wertebereich von 0..100%. Im Eigenschaftsdialog auf der Seite Fortschrittsbalken ist anzugeben welchem Variablenwert 0% und welchem 100% entsprechen. Weiter lässt sich der Balken in alle vier Richtungen konfigurieren und mit einer Skala versehen.

