# Embedded Webserver für FP Gateways
# Handbuch


Version: 2.2.1



www.inovolabs.com

Redaktionsschluss: 23.02.2021

# Inhaltsverzeichnis

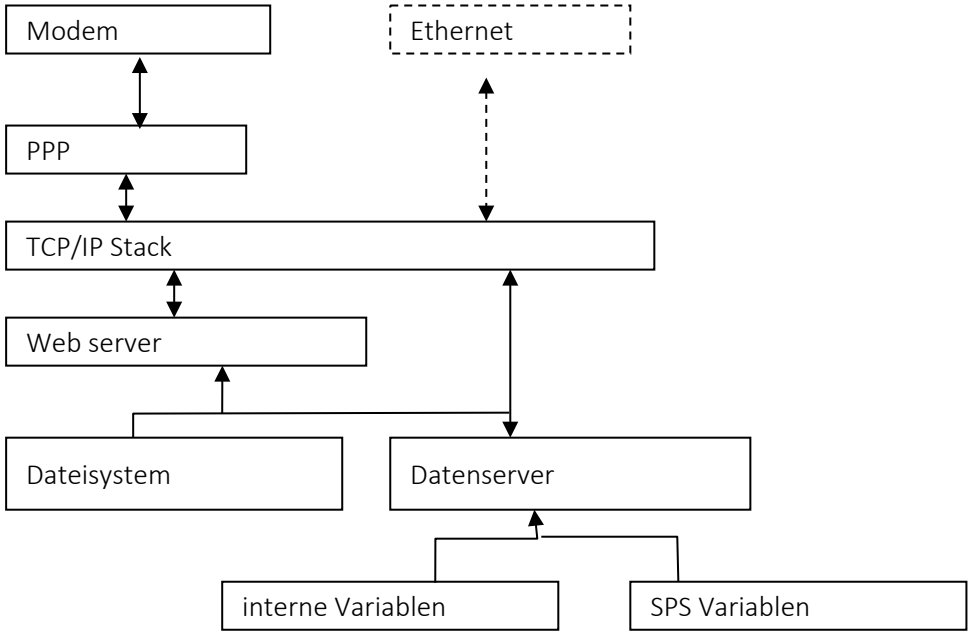
# 1 Webserver

Die FP IoT Gateways sind mit einem eingebauten Webserver verfügbar, welcher über eine PPP-Einwahl oder Ethernetverbindung die Darstellung von SPS-Daten in einem Webbrowser erlaubt.

**Webserver Zugriff**
Standard IP-Adresse: 192.168.0.1.
Startseite: index.html.

Das FP IoT Gateway kann bis zu acht http- oder TiXML-Verbindungen via TCP/IP gleichzeitig öffnen.

Der Webserver kann auch über ein Dial-Out spezieller ISPs angesprochen werden.




Der Webserver zeigt im Dateisystem abgelegte HTML-Seiten an. Die Seiten können Referenzen zu internen oder SPS-Variablen beinhalten, welche vom Datenserver ausgeschlüsselt werden. Dies wird über spezielle HTML-Codes erreicht, welche den Istwert der Variable beim Öffnen der Webseite einfügen. Ein direkter Zugriff auf den Datenserver ist via CGI-Skript möglich, um somit dynamisch aktualisierte Webseiten zu erstellen. Der Webserver unterstützt einen HTTP Dateicache.

Auf http://www.inovolabs.com erhalten Sie HTML-Beispiele sowie eine Bibliothek von Webseitenelementen.

Zusätzliche Informationen über den Datenzugriff via HTTP erhalten Sie im Handbuch „SCADA / GLT Anbindung für FP Gateways“.

## 1.1 Anzeige von SPS- und Gateway-Variablen

Um die Werte von Variablen über eine Webseite darzustellen, muss eine Referenz auf deren Wert in den HTML-Code eingefügt werden. Die in HTML zu verwendende Referenz ist identisch zur TiXML-Referenz: Sie beginnt mit dem Referenz-Code (® oder in XML: ®), gefolgt vom Pfad der Variable ( z.B.: /Process/MB/IO/I/P0 um den Eingang eines Hutline-Modems anzuzeigen) .

Teil einer Webseite, die Eingänge anzeigt:

```
<TABLE ID="Table1" BORDER=1 CELLSPACING=3 CELLPADDING=1 WIDTH=200>
  <TR>
    <TD WIDTH=104><P>Port P0:</TD>
    <TD WIDTH=77>&#xae;/Process/MB/IO/I/P0;</TD>
  </TR>
  <TR>
    <TD WIDTH=104><P>Port P1:</TD>
    <TD WIDTH=77>&#174;/Process/MB/IO/I/P1;</TD>
  </TR>
</TABLE>
```

## 1.2 Spezielle Techniken zum Erzeugen dynamischer Webseiten

### 1.2.1 CGIs

Der Datenserver kann direkt via CGI-Skript angesprochen werden und unterstützt Logfilezugriff sowie Zugriff auf einzelne oder Gruppen von Variablen.

| | |
|---|---|
| `cgi-bin/readLog.exe?` | zum Logfile Lesen |
| `cgi-bin/readVal.exe?` | zum Variablen Lesen |
| `cgi-bin/writeVal.exe?` | zum Variablen Schreiben |
| `cgi-bin/RefreshValues.exe?` | zum Variablengruppen Aktualisieren |
| `cgi-bin/tixml.exe?` | zum Verarbeiten von TiXML Kommandos über http POST |

Die CGIs können in HTML Forms via „GET“ verwendet werden (Ausnahme: tixml-exe).
Die Syntax einiger dieser Befehle ist identisch zu den Befehlen der „Ininet SpiderControl“ Software. Es ist somit möglich „SpiderControl“ TEQ-Dateien in die Webseiten einzubauen. Diese TEQ Dateien enthalten eine Anzeige, die die Werte periodisch über den Datenserver aktualisiert. Für weitere Informationen ziehen Sie bitte das SpiderControl Handbuch zu Rate. Verwenden Sie dabei die oben angegebenen Referenzen als PPOs.

*Beispiele*

(weitere Beispiele können auf http://www.inovolabs.com gefunden werden)

*Beispiel 1*

Dieser Code erzeugt einen Button zum Abruf der letzten 10 Werte eines Logfiles. Das Logfile wird im gleichen Fenster angezeigt.
Der „Range“ wird nach dem + angegeben. Es werden die vom gleichnamigen TiXML-Befehl bekannten Bereiche unterstützt, jedoch müssen die Zeichen „#“ durch „%23“ sowie „-“ durch „%2D“ ersetzt werden.

```
<html>
 <body>
    <input type=button
    onclick="location='cgi-bin/readLog.exe?Event+%2310%2D;" value="Event-Log
    lesen">
 </body>
</html>
```

*Beispiel 2*

Dieser Code erzeugt einen Button zum Setzen vom Ausgang Q2 (Alias, siehe nächstes Kapitel). Das Ergebnis wird im selben Fenster angezeigt:

```
<html>
 <body>
   <input type=button value="set Q/P2"
   onclick="location='cgi-bin/writeVal.exe?QP2+1';" >
</body>
</html>
```

*Beispiel 3*

Dieser Code erzeugt ein Eingabefeld zum Schalten von QP2. Das Ergebnis wird mittels Javaskript in einem zweiten Fenster angezeigt:

```
<html>
<script language="JavaScript">
function window()
{
NewWindow1=window.open("","Result","width=20,height=20,resizable=no,scrollb
ars=no,top=50,left=50");
NewWindow1.focus();
}
</script>
<body>
 <FORM action="../cgi-bin/writeVal.exe" method="get" target="Result">
    Q/P2 Value:
    <INPUT type="text" name="QP2" value="">
    <INPUT type="submit" value="set" onclick="window()">
 </FORM>
</body>
</html>
```

### 1.2.2 HTML Befehle

Das FP IoT Gateway und sein Datenserver kann über spezielle HTML-Befehle angesprochen werden:

| HTML Befehle | | |
|---|---|---|
| *Syntax* | ®(*commandname=parameters*); | |
| *Beschreibung* | HTML Befehl zur Steuerung des FP IoT Gateways Webservers | |
| *Parameter* | Disconnect | Das FP IoT Gateway trennt die Verbindung. |
| | Format | Zeigt eine Variable und formatiert den Wert, z.B. ®(Format=/Process/MB/IO/I/P0+?on'off);<br>Die Formatanweisung hinter dem „+“ ist identisch zur „External“ oder „Record“ Formatanweisung (siehe TiXML Reference Manual). Beachten Sie, dass die Einheit per Hochkomma „ '“ statt Semikolon „;“ angegeben wird. Bei boolschen Formaten wird das trennende Komma „,“ ebenfalls durch ein Hochkomma „ '“ ersetzt. Das Format ist nur bei Direktreferenzen, also nicht bei Alias-Variablen möglich. |
| | Replace | Zeigt eine Variable und ersetzt einen Teil des Inhalts. z.B.: ®(Replace=/Process/PV/Text:String_to_replace:new_string);<br>Der Wert der referenzierten Variable wird nach dem String „String_to_replace“ durchsucht, der durch „new_string“ ersetzt wird. |

| | | |
|---|---|---|
| | Include | Fügt den „Process“ Zweig oder einen Teilbereich in das HTML-Dokument ein. |
| | ReadLog | Zeigt den Inhalt eines Logfiles als HTML-Tabelle. |

*Beispiele*

*Beispiel 1*

Wenn diese Seite geladen wird, trennt das FP IoT Gateway die Verbindung:

```
<HTML>
  <BODY>
       Disconnecting... &#xae;(Disconnect);
  </BODY>
</HTML>
```

*Beispiel 2*

Der Wert „12345“ der Variable Word02 sowie der Wert 0/1 der Variable Bit02 wird formatiert dargestellt:

```
<HTML>
  <BODY>
   &#xae;(Format=/Process/Bus1/Device_0/Word02+F,2'm);<br>
    &#xae;(Format=/Process/Bus1/Device_0/Bit02+?on’off);
  </BODY>
</HTML>
```

Angezeigter Wert: 123,45m

*Beispiel 3*

Die Zeichenkette „Company Name“ der Variable „Text“ wird bearbeitet:

```
<HTML>
  <BODY>
    &xae;(Replace=/Process/PV/Text:Name:Tixi);
  </BODY>
</HTML>
```

Angezeigte Zeichenkette: „Company Tixi“

*Beispiel 4*

Die Hutline I/Os aus dem „Process“-Zweig werden in der Webseite angezeigt:

```
<HTML>
  <BODY>
       &#xae;(Include=/Process/MB/IO/);
  </BODY>
</HTML>
```

*Beispiel 5*

Dieses Beispiel zeigt den Inhalt der letzten 30 Minuten des Datalog-Logfiles direkt als HTML-Tabelle an:

```
<html>
 <body>
   <div align=”center”>
       <p><u>Logdata of last 30 minutes:</u></p>
       <table cellpadding=”5” cellspacing=”1” border=”1”
       align=”center”>
           &#xae;(ReadLog=Datalog+last 30 minutes);
       <table>
   </div>
 </body>
</html>
```

### 1.2.3 Alias Namen für Variablenreferenzen

Für vereinfachtes Programmieren von CGI, Applet und HTML-Code bietet das FP IoT Gateway eine Datenbank mit Aliasnamen für I/O und SPS-Variablenreferenzen. Dadurch ist es möglich innerhalb des Source-Codes statische Namen zu verwenden, und diese im TiXML-Projekt mit den Variablen zu verknüpfen.

Um den Wert eines Eingangs ohne Alias Datenbank zu erhalten, ist folgende Anweisung nötig:
HTML-Referenz: `&#xae;/Process/MB/IO/I/P0;`
CGI Aufruf: `cgi-bin/readVal.exe?/Process/MB/IO/I/P0`
Die Moduladresse MB ist in der Referenz enthalten. Um die Referenz mit einem anderen Modul zu verwenden, z.B. Adresse C42, müsste der Code verändert werden.

Mit der Aliasdatenbank ist es nun möglich einen statischen Namen, z.B. „`ExtIP0`“ auf eine Referenz `"/Process/MB/IO/I/P0"` zu leiten. Der statische Name wird innerhalb der Referenz verwendet:
HTML: `&#xae;ExtIP0`
CGI Aufruf: `cgi-bin/readVal.exe?ExtIP0`
und vom System durch den Wert der Referenz ersetzt. Das ermöglicht es, die gleiche Webseite mit unterschiedlicher Hardware lediglich durch Änderung der Alias Datenbank zu verwenden.

Ein weiterer Vorteil der Alias Datenbank ist die Möglichkeit Zugriffsrechte auf Variablen festzulegen. Dadurch kann der Schreibzugriff auf Variablen verhindert werden.

Zusätzlich kann die Ausgabe der Variablen formatiert werden. Wenn z.B. in der External Definition eine andere Formatierung verwendet wird, als im Webserver gewünscht, kann diese über die Alias Datenbank separat festgelegt werden. Die Formatanweisungen sind identisch zu denen der Logfile-Records (siehe TiXML Reference Manual) oder SPS Variablen.
Datenbank Pfad: /ISP/WebServer

```
<WebServer>
   <Aliases>
     <ExtIP0 _="/Process/MB/IO/I/P0" acc="R" />
     <ExtOP0 _="/Process/MB/IO/Q/P0" acc="RW" />
     <ExtOP1 _="/Process/MB/IO/Q/P1" acc="RW" format="?on,off"/>
     <PLCI1 _="/Process/Bus1/D0/I1" acc="R" format="F;"/>
     <PLCO1 _="/Process/Bus1/D0/O1" acc="RW" />
   </Aliases>
</WebServer>
```

## 1.3 LogAliases zur Formatierung von Logdaten

Logdateien werden vom FP IoT Gateway in XML-Form ausgegeben.
Mit der „LogAliases“ Gruppe ist es möglich die Ausgabe zu formatieren, um z.B. CSV-Dateien zu erzeugen.

Die „LogAliases“ Gruppe ist Teil der ISP/WebServer Gruppe:

Datenbank Pfad: /ISP/Webserver

```
<WebServer>
   <LogAliases>
     <Aliasname _="Logfile" FORMAT saveAs=”Dateiname”/>
   </LogAliases>
…
</WebServer>
```

Die FORMAT Anweisungen sind identisch zu denen des „IncludeLogTXT“ Befehls, beschrieben im Kapitel Datenlogging des TiXML Reference Manuals.
Der bei „saveAs“ (ab FW 2.2) angegebene Dateiname wird beim Speichern der Datei vorgegeben (ab Microsoft Internet Explorer 6.0.2900)

*Beispiel*
Datenbank Pfad: /ISP/WebServer

```
<WebServer>
    <LogAliases>
        <DatalogCSV _="Datalog" type="CSV" saveAs=“Datalog.csv“/>
    </LogAliases>
…
</WebServer>
```

## 1.4 Hochladen einer Website

Der komplette Verzeichnisstruktur der Webseite muss in eine BASE64 webSrc.bin Datei kompiliert werden. Die webSrc.bin muss in einen TiXML-Rahmen verpackt werden:

```
[<SetBinary _="HTTP" mode="Base64" name="WebSrc.bin">
<D _="NgAAAPYHAADYAycUBAElNYXN0ZXIuamFyACYcAABpbmRleC5odG1sAKcIAABJT19Q"/>
...
<D _="CiAgICAgICAgPUj4NCiAgICA8L1RBQkxFPg0KPC9CT0RZPg0KPC9IVE1MPg0KIA=="/>
</SetBinary>]
```

Tixi.Com stellt ein Tools zur Konvertierung in ASCII-Dateien zur Verfügung.

## 1.5 TFTP Dateitransfer

Um Dateien, z.B. Webseiten zum oder vom FP IoT Gateway zu übertragen, kann ein TFTP-Programm verwendet werden. Das FP IoT Gateway ermöglicht diese Kommunikation über eine aktive TCP/IP Verbindung (Ethernet oder Internet).
Kostenlose TFTP Programme sind im WWW erhältlich.

Die ISP Datenbank wird verwendet um die Binärdateien und den Zugriff auf die Dateien zu konfigurieren.

Datenbank Pfad: /ISP/TFTP

| TFTP – Dateitransfer | | |
|---|---|---|
| *Syntax* | <TFTP><br><Port _="*Number*"/><br><Files><br><*Description* _="*filename*" acc="*access*" size="*size*"/><br></Files><br></TFTP> | |
| *Beschreibung* | Datei Definition für TFTP Dateitransfer. | |
| *Parameter* | *Number:* | TCP/IP Port der TFTP Kommunikation. Standard: 69 |
| | *Description:* | Beschreibung der verfügbaren Dateien. |
| | *filename:* | Name der registrierten Datei |
| | *access:* | Zugriffsrechte für diese Datei:<br>R – nur Lesezugriff |

| | |
|---|---|
| | W – nur Schreibzugriff<br>RW – Lese- und Schreibzugriff<br>*Size:* Maximal mögliche Größe in Byte (während Upload) |
| *Beispiel* | Drei Dateien sind angelegt: Die Tixi FP IoT Gateway<br>- Webseite als Binärdatei<br>- eine Kopie des TiXML-Projekts als Zip-Datei<br>- eine weitere Binärdatei mit Grafiken für die Webseite:<br><br>`<TFTP>`<br>`  <Port _="69"/>`<br>`  <Files>`<br>`  <Website _="websrc.bin" acc="RW" size="40960"/>`<br>`  <Project _="project.zip" acc="R" size="40960"/>`<br>`  <webpictures _="pictures.bin" acc="RW" size="10240"/>`<br>`  </Files>`<br>`</TFTP>` |

## 1.6 Webserver Konfiguration

Die ISP Datenbank beinhaltet weitere Webservereinstellungen. Dies ist die Standardkonfiguration:

Datenbank Pfad: /ISP/Webserver

```
<WebServer>
    <ActiveSite _="Site1"/>
    <Site1>
        <Archive1 _="WebSrc.bin"/>
    </Site1>
    <Site2>
        <Archive1 _=""/>
        <Archive2 _=""/>
        <Archive3 _=""/>
    </Site2>
</WebServer>
```

### 1.6.1 Mehrfache Webseiten, Webseiten aufteilen

Es ist möglich mehrere Webseiten im FP IoT Gateway abzulegen. Die aktive Seite wird über „ActiveSite“ festgelegt. Das FP IoT Gateway kann den Webinhalt auf bis zu drei Archive verteilen. Damit ist es möglich, oft geänderte Dateien von selten geänderten zu trennen, um Übertragungszeit zu sparen.

| Webserver - Sites | |
|---|---|
| *Syntax* | <WebServer><br>  <ActiveSite _="*SiteName*"/><br>  <SiteName1><br>  <Archive1 _="FileName"/><br>  </SiteName1><br>  <SiteName2><br>  <Archive1 _="FileName"/><br>  <Archive2 _="FileName"/><br>  <Archive3 _="FileName"/><br>  </SiteName2><br></WebServer> |
| *Beschreibung* | Site Konfiguration des Webservers. |
| *Parameter* | *ActiveSiteName:* Name der aktiven Site.<br>*SiteNameX:* Namen der verfügbaren Sites.<br>*FileName:* Namen der Binärdateien mit Webinhalt. |
| *Beispiel* | Zwei Sites sind konfiguriert und Site2 ist aktiv. Die Webseite ist in drei Teile aufgeteilt. Einer mit dem HTML Code, einer mit Bildern und einer mit Java-Applets:<br><br><WebServer><br>  <ActiveSite _="Site2"/><br>  <Site1><br>  <Archive1 _="WebSrc.bin"/><br>  </Site1><br>  <Site2><br>  <Archive1 _="html.bin"/><br>  <Archive2 _="pictures.bin"/><br>  <Archive3 _="applets.bin"/><br>  </Site2><br></WebServer> |

### 1.6.2 Webserver Verzeichnis-Zugriffsbeschränkungen

Für jede FP IoT Gateway Webserver „Site“ können einzelne Verzeichnisse oder Dateien im Zugriff beschränkt werden (Realms). Die notwendige Authentifizierung ist für alle Seiten innerhalb des HTML-Pfads gültig, und wird aus der AccessRights Datenbank übernommen. (siehe TiXML Reference Manual).

| Eingeschränkter Verzeichniszugriff | |
|---|---|
| *Syntax* | `<Site1>`<br>`  <Archive1 _="WebSrc.bin"/>`<br>`  <Restricted>`<br>`  <Restriction Path="path" Realm="name" AccLevel="level"/>`<br>`  </Restricted>`<br>`</Site1>` |
| *Beschreibung* | Site Konfiguration des Webservers. |
| *Parameter* | *Restriction:* Name der Beschränkung.<br>*path:* Relative URL des beschränkten Verzeichnisses<br>*name:* frei wählbarer Name des Zugriffsbereichs<br>*level:* Access Level für diese Beschränkung (siehe TiXML Reference Manual) |
| *Beispiel* | Eine Site ist angelegt. Zugriff zum SYSTEM/ Pfad ist nur für Benutzer mit Access Level 2 gestattet. Alle anderen Verzeichnisse (*) sind für alle offen.<br><br>Database path: /ISP/WebServer<br><br>`<WebServer>`<br>`  <ActiveSite _="Site1"/>`<br>`  <Site1>`<br>`  <Archive1 _="WebSrc.bin"/>`<br>`  <Restricted>`<br>`  <System Path="SYSTEM/*" Realm="TAM_SYSTEM"`<br>`  AccLevel="2"/>`<br>`  <DeviceData Path="*" Realm="TAM_ALL"`<br>`  AccLevel="1"/>`<br>`  </Restricted>`<br>`  </Site1>`<br>`</WebServer>` |

### 1.6.3 Webserver Verbindungstimeout

Nach 60 s Inaktivität trennt das FP IoT Gateway die Verbindung automatisch. Wenn diese Zeit erhöht oder verkürzt werden soll, muss das Timeout für ein- und ausgehende Verbindungen getrennt definiert werden.

Für ausgehende Verbindungen, z.B. CBIS, kann folgender Eintrag zur Webserverkonfiguration hinzugefügt werden:

| Webserver – Verbindungstimeout | |
|---|---|
| *Syntax* | **`<KeepConnected _="periode"/>`** |
| *Beschreibung* | Zeit der Inaktivität, nach der das Modem auflegt (nur bei ausgehenden Verbindungen, z.B. CBIS). |
| *Parameter* | *periode* = Zeit (z.B. 60s, 5m, 1h,…) |

*Beispiel*

Trennung nach 5 Minuten:

Datenbank Pfad: /ISP/WebServer

```
<WebServer>
    <KeepConnected _=”300s”/>
    <ActiveSite _="Site1"/>
    <Site1>
        …
    </Site1>
    <Site2>
        …
    </Site2>
</WebServer>
```

Für eingehende Verbindungen muss ein Eintrag in der Timeout-Datenbank vorgenommen werden:

```
[<SetConfig _="USER" ver="v">
    <Timeouts>
        <PPPTimeout _="600s"/>
    </Timeouts>
</SetConfig>]
```

### 1.6.4 Webserver TCP/IP Port

Der FP IoT Gateway Webserver nimmt HTTP-Anfragen auf Port 80 entgegen. Wenn das Modem hinter einer Firewall angeschlossen ist, kann der HTTP-Port auch geändert werden:

| Webserver – TCP/IP port | |
|---|---|
| *Syntax* | **<Port _=”*number*”/>** |
| *Beschreibung* | TCP/IP Port zur Verbindung zum Webserver. |
| *Parameter* | *number* = Portnummer, z.B. 8080 (Standard: 80) |

*Beispiel*

Verbindung auf Port 8080:

Datenbank Pfad: /ISP/WebServer

```
<WebServer>
    <Port _=”8080”/>
    <ActiveSite _="Site1"/>
    <Site1>
        …
    </Site1>
    <Site2>
        …
    </Site2>
</WebServer>
```

### 1.6.5 Webserver IP-Adressbereich beschränken

Der Adressbereich des Webservers der FP IoT Gateways kann auf einen bestimmten Bereich beschränkt werden. Damit kann sichergestellt werden, dass nur IP-Adressen aus einem bestimmten IP-Adressbereich Zugriff auf das IoT Gateway erhalten.
Die Zugriffsbeschränkung kann auch dafür verwendet werden, externe Zugriff auf den Webserver komplett zu verhindern.

| Webserver – Adressbereich einschränken | |
|---|---|
| *Syntax* | **<RestrictionStartIP _="*Start IP*" />**<br>**<RestrictionEndIP _="*End IP*" />** |
| *Beschreibung* | Adressbereich zur Einschränkung des Webserver-Zugriffes<br>Beide Optionen müssen immer gemeinsam verwendet werden.<br>Um den Webserver-Zugriff von außen komplett zu unterbinden, müssen die *Start IP* und die *End IP* auf den Wert `127.0.0.1` gesetzt werden. |
| *Parameter* | *Start IP* = Start-IP-Adresse, ab der ein Zugriff erlaubt werden soll<br>*End IP* = End-IP-Adresse, bis zu der ein Zugriff erlaubt werden soll |
| *Beispiel 1* | Adressbereich auf `192.168.0.100` bis `192.168.0.200` einschränken:<br><br>Datenbank Pfad: /ISP/WebServer<br><br>`<WebServer>`<br>`  <Port _="80"/>`<br>`  <RestrictionStartIP _="192.168.0.100"/>`<br>`  <RestrictionEndIP _="192.168.0.200"/>`<br>`  <ActiveSite _="Site1"/>`<br>`  <Site1>`<br>`  …`<br>`  </Site1>`<br>`  <Site2>`<br>`  …`<br>`  </Site2>`<br>`</WebServer>` |
| *Beispiel 2* | Zugriff auf Webserver komplett sperren:<br><br>Datenbank Pfad: /ISP/WebServer<br><br>`<WebServer>`<br>`  <Port _="80"/>`<br>`  <RestrictionStartIP _="127.0.0.1"/>`<br>`  <RestrictionEndIP _="127.0.0.1"/>`<br>`  <ActiveSite _="Site1"/>`<br>`  <Site1>`<br>`  …`<br>`  </Site1>`<br>`  <Site2>`<br>`  …`<br>`  </Site2>`<br>`</WebServer>` |

### 1.6.6 TiXML TCP/IP Port

Das Tixi FP IoT Gateway kann TiXML-Befehle über TCP/IP empfangen und verarbeiten. Der TCP/IP Port der Kommunikation wird in der Gruppe „TiXML“ der Datenbank ISP festgelegt.

Datenbank Pfad: /ISP/TiXML

| TiXML – TCP/IP Port | |
|---|---|
| *Syntax* | **<Port _=”*number*”/>** |
| *Beschreibung* | TCP/IP Port für TiXML Kommunikation. |
| *Parameter* | *number* = Portnummer, z.B. 8300 (standard) |
| *Beispiel* | Verbindung über Port 8300:<br><br><TiXML><br><Port _="8300"/><br></TiXML> |

Um mit dem FP IoT Gateway über das Netzwerk zu kommunizieren, kann sowohl TICO – TiXML Console als auch jedes Terminal-Programm mit TCP/IP Unterstützung verwendet werden (z.B. HyperTerminal, Telnet) oder ein virtueller serieller Port via TCP/IP (z.B. Tibbo).

| TiXML – Zugriff über Adressbereich beschränken | |
|---|---|
| *Syntax* | **<RestrictionStartIP _="*Start IP*" /><br><RestrictionEndIP _="*End IP*" />** |
| *Beschreibung* | Adressbereich zur Einschränkung des TiXML-Zugriffes über TiXML-Port.<br>Beide Optionen müssen immer gemeinsam verwendet werden.<br>Um den TiXML-Zugriff von aussen komplett zu unterbinden, müssen die *Start IP* und die *End IP* auf den Wert `127.0.0.1` gesetzt werden. |
| *Parameter* | *Start IP* = Start-IP-Adresse, ab der ein Zugriff erlaubt werden soll<br>*End IP* = End-IP-Adresse, bis zu der ein Zugriff erlaubt werden soll |
| *Beispiel* | Zugriff auf Webserver komplett sperren:<br><br><TiXML><br><Port _="8300"/><br><RestrictionStartIP _="127.0.0.1"/><br><RestrictionEndIP _="127.0.0.1"/><br></TiXML> |

## 1.7 Webserver Verbindungseinstellungen

### 1.7.1 Dial-in Zugriff

Damit ein PPP-Client, z.B. das Windows-DFÜ-Netzwerk, eine Verbindung zum FP IoT Gateway herstellen kann, muss ein PPP-Server aktiviert werden. In diesem werden die TCP/IP-Daten der Punkt-zu-Punkt-Verbindung festgelegt.
Der Zugriff über ISDN syncPPP Protokoll ist nur möglich, wenn die Rufnummer des Anrufenden zuvor im Modem hinterlegt wurde. Hiernach ist eine Fernkonfiguration des Modems von diesem Anschluss nur noch über X.75 möglich. Dieser Eintrag ist für Analogmodems unrelevant.
Die Konfiguration wird in der Gruppe „PPP_Server“ der ISP Datenbank vorgenommen:

Datenbank Pfad: /ISP/PPP_Server

| Webserver – PPP Zugriff | |
|---|---|
| *Syntax* | **<PPP_Server><br><OwnIP _="ServerIP"/><br><OwnSubnet _="SubnetMask"/><br><RemoteIP _="ClientIP"/><br><AuthentFlags _="Flag"/><br><DOPx _="*CallerID* "/><br></PPP_Server>** |
| *Beschreibung* | Konfiguration des PPP-Servers. |
| *Parameter* | *ServerIP:* IP-Adresse des PPP-Servers (Modem)<br>*SubnetMask:* Subnetzmaske des verwendeten IP-Netzes<br>*ClientIP:* IP-Adresse, die dem PPP-Client (PC) zugewiesen wird<br>*Flag:* Authentifizierungsmethode bei der PPP-Einwahl<br>1 = PAP<br>2 = CHAP<br>3 = AUTO<br>*x*: Nummer des DOP-Eintrags (hochzählen)<br>*CallerID:* Rufnummer des eingehenden Anrufs |
| *Beispiel* | Das Modem verwendet die private IP-Adresse 192.168.0.1 (standard) und weist dem eingewählten Gerät die IP 192.168.0.10 zu.<br>Das Gerät mit der Rufnummer 0301234567 kann sich via ISDN-syncPPP einwählen.<br><br><PPP_Server><br><OwnIP _="192.168.0.1"/><br><OwnSubnet _="255.255.255.0"/><br><RemoteIP _="192.168.0.10"/><br><AuthentFlags _="3"/><br><DOP1 _="0301234567"/><br><DOP2 _=""/><br><DOP3 _=""/><br></PPP_Server> |

### 1.7.2 Call back initiation service CBIS (nur bei Geräten der 5.Generation)

Eine weitere Möglichkeit auf den FP IoT Gateway Webserver zuzugreifen ist der „call back initiation service“ (CBIS). Dieser Service nutzt eine Einwahl über Telefonleitungen und steht nur bei der älteren Gerätegeneration „G5“ zur Verfügung, also z.B. bei Geräten der Serie H400. In der aktuellen Generation „G6“ und in zukünftigen Generationen ist der CBIS-Service nicht mehr verfügbar, weil die Technik der Telefoneinwahl nicht mehr zeitgemäß ist und von vielen Serviceprovidern nicht mehr unterstützt wird.
Wenn die CBIS Prozedur gestartet wird, was über einen Anruf oder EventHandler (siehe TiXML Reference Manual) geschehen kann, wählt sich das FP IoT Gateway ins Internet ein und sendet eine E-Mail mit der aktuellen IP-Adresse an den voreingestellten Empfänger.

Die CBIS Funktion benötigt einen Internetzugang (siehe TiXML Reference Manual) und ist Teil der ISP Datenbank.

Datenbank Pfad: /ISP/CBIS

| CBIS – Call back initiation service | |
|---|---|
| *Syntax* | **<CBIS><br><ServerName _="*Address*"/><br><PhoneNumber _="*CallerID*"/>** |

| | |
|---|---|
| | **<Account _="*Recipient*"/>**<br>**<ResponseTime _="*IdleTime*"/>**<br>**</CBIS>** |
| *Beschreibung* | Konfiguration des Call back initiation service. |
| *Parameter* | *Address*: Mailserver<br>*CallerID:* CallerID des Anrufenden<br>*Recipient:* Empfänger der CBIS E-Mail<br>*IdleTime:* Wartezeit bis zum ersten Zugriff (connect) |
| *Beispiel*  | Ein Internetzugang via „Freenet“ ist konfiguriert. Die CBIS E-Mail wird über den Server „193.101.167.194“ an den Empfänger cbis@devicecontrolweb.com geschickt. Die CBIS Prozedur wird über einen Anruf von der Nummer „0301234567“ ausgelöst:<br><br>`<CBIS>`<br>`<ServerName _="193.101.167.194"/>`<br>`<PhoneNumber _="0301234567"/>`<br>`<Account _="cbis@devicecontrolweb.com "/>`<br>`<ResponseTime _="60s"/>`<br>`</CBIS>` |

Der CBIS CallerID-Auslöser benötigt einen speziellen System-EventHandler mit dem CBIS Befehl:

Datenbank Pfad: /EVENTS/EventHandler/System

```
<System>
    <CBISRequest>
     <CBIS/>
    </CBISRequest>
</System>
```

Die CBIS Prozedur kann über jeden EventHandler mit dem CBIS Befehl gestartet werden (siehe TiXML Reference Manual).

Als Ergebnis der CBIS Prozedur erhält der Empfänger „Account“ eine E-Mail vom Absender CBIS@Tixi.Com mit folgendem Betreff:

```
Betreff: CBIS Connect IP-Address   (z.B. CBIS Connect 192.168.0.1)
```

Der Nachrichtentext beinhaltet den SITE_TAG (siehe „Tixi HTTP Data Interface“ Handbuch).

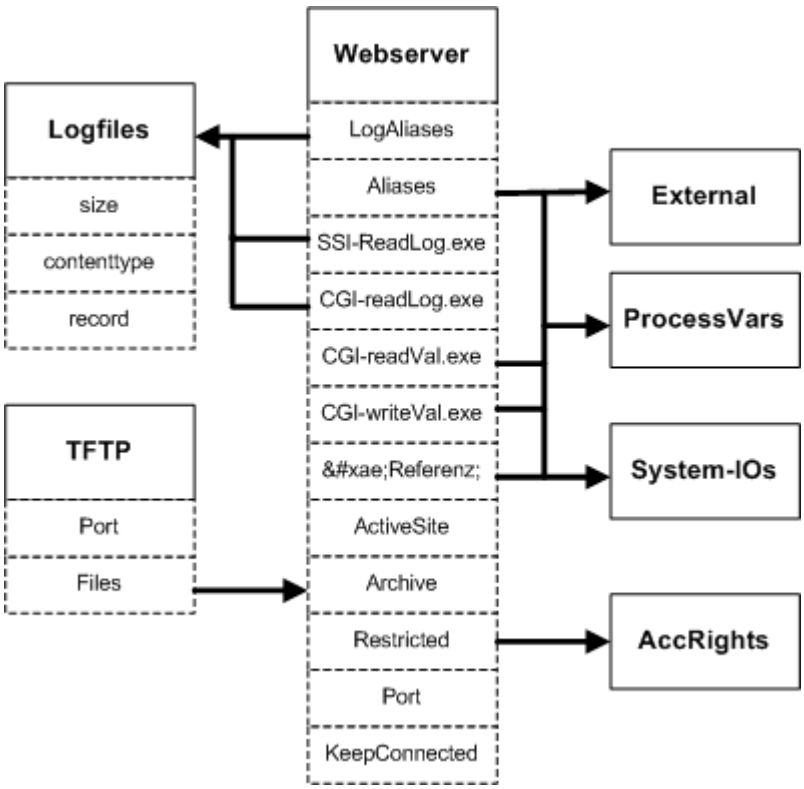
Danach hat der Empfänger die bei „ResponseTime“ angegebene Zeit, um auf den Webserver zuzugreifen.

### 1.7.2.1 CBIS Absenderadresse ändern

Einige Mailserver erlauben den Nachrichtenversand mit der Adresse CBIS@Tixi.Com nicht.
In diesem Fall kann ein MessageJobTemplate (siehe TiXML Reference Manual) erzeugt werden, bei dem der Absender und Empfänger aus dem Adressbuch gewählt werden kann:

```
<SendIP _="CBIS">
  <Sender _="/D/AddressBook/MySelf"/>
  <Recipient _="/D/AddressBook/Receiver_0"/>
..
..
</SendIP >
```

# 2 Anhang: Projektstruktur und –zusammenhänge



# Notizen