

# Das JUNG Facility-Pilot System

## Version 2.0

### Inhalt:

<b>1 EINFÜHRUNG.....</b>	<b>2</b>
<b>2 NEU IN JUNG FACILITY-PILOT VERSION 2.0.....</b>	<b>3</b>
2.1 EIB EDITOR.....	3
2.2 VISUALISIERUNG.....	3
2.3 PROZESSMODELLE.....	4
2.4 OPC EDITOR.....	5
2.5 MUSIKSTEUERUNG.....	6
<b>3 SCHWERPUNKTE.....</b>	<b>7</b>
3.1 EINFACHHEIT UND KOMFORT.....	7
3.2 FLEXIBILITÄT FÜR WECHSELNDE ANFORDERUNGEN.....	7
<b>4 TECHNISCHE VORAUSSETZUNGEN.....</b>	<b>8</b>
4.1 HARDWARE.....	8
4.2 BETRIEBSSYSTEME.....	8
4.3 PROZESSANSCHLUSS.....	8
4.4 IMPORT VON DATEN AUS DER ETS.....	8
4.5 MICROSOFT INTERNET EXPLORER, VERSION 6; DIRECTX, VERSION 9.0c.....	8
<b>5 LOGISCHER AUFBAU DES GESAMTSYSTEMS.....</b>	<b>9</b>
5.1 EIB- UND OPC- PROZESSANSCHLUSS.....	10
5.2 PROZESSMODELLE.....	10
5.3 KALENDER.....	10
5.4 VISUALISIERUNG.....	10
5.5 SYSTEMSTEUERUNG.....	10
<b>6 BETRIEBSARTEN.....</b>	<b>11</b>
6.1 ALS DEMO- VERSION:.....	11
6.2 ALS AKTIVIERTE PRODUKTVERSION:.....	11
6.3 ALS NICHT- AKTIVIERTE PRODUKTVERSION:.....	11
<b>7 KOPIERSCHUTZ UND AKTIVIERUNG.....</b>	<b>11</b>
<b>8 ABLAGE VON PROJEKTEN.....</b>	<b>12</b>
<b>9 DATENSICHERUNG UND NOTBETRIEB.....</b>	<b>12</b>
<b>10 ANSCHRIFT.....</b>	<b>13</b>

# 1 Einführung



Das **JUNG Facility-Pilot- System** ist ein Paket aus WINDOWS - Applikationen mit einheitlicher Bedienoberfläche zur Visualisierung, Steuerung und Überwachung technischer Prozesse, insbesondere in der Gebäudeautomatisierung.

Das System bietet für einen Benutzer die Möglichkeit, durch Prozessbilder einen Überblick über die technischen Prozesse zu erhalten und steuernd einzugreifen, Daten zu archivieren und anzuzeigen, Störungen und Alarme zu definieren, und die Signalisierung von Alarmen zu konfigurieren.

Das System kann zugleich als automatische Steuerung verwendet werden, indem Reaktionen auf Ereignisse festgelegt werden. Dies geschieht in der Art von ‚virtuellen Geräten‘, die im System erzeugt und einfach konfiguriert werden, unter Nutzung von einfach und flexibel konfigurierbaren Kalendern, bei Bedarf auch ergänzt durch Scripts.

Das JUNG Facility-Pilot- System kann mit OPC- Servern für unterschiedliche Prozessanschlüsse verwendet werden, enthält aber auch einen direkten Anschluss an den European Installation Bus (EIB). In diesem Zusammenhang verfügt das JUNG Facility-Pilot- System auch über Komponenten zur Inbetriebnahme von EIB- Systemen und zur Protokollierung des Datenverkehrs.

Weitere Funktionen betreffen zum Beispiel die Benutzerverwaltung, das Logbuch und die Alarmbehandlung.

Insgesamt steht bei dieser Funktionsvielfalt aber immer die einfache und komfortable Bedienung im Vordergrund.

## **2 Neu in JUNG Facility-Pilot Version 2.0**

Die JUNG Facility-Pilot Version 2.0 erweitert erheblich die Einsatzmöglichkeiten des JUNG Facility-Pilot – Systems.

Beim Einsatz der neuen Version für bereits vorhandene Projekte fordern wir unsere Kunden aber auf, genau die Auswirkungen zu untersuchen, die sich insbesondere aus den Änderungen des Prozessmodells für die konkreten Projekte ergeben.

Außerdem werden ältere WINDOWS – Versionen vor WINDOWS XP ab Version 2.0 des JUNG Facility-Pilot – Systems nicht mehr unterstützt.

### **2.1 EIB Editor**

Der EIB Editor ist nun in Verbindung mit dem Falcon - Treiber ab Version 1.3 für die Verwendung mit EIBnet/IP vorbereitet.

Leider ist zur Zeit noch nicht klar, ob für die Verwendung mit EIBnet/IP eine zusätzliche Lizenz der EIBA erforderlich ist, die wir in diesem Fall leider Kunden in Rechnung stellen müssen, die EIBnet/IP einsetzen möchten. Wir hoffen, dass wir den EIBnet/IP – Anschluss des FALCON-Treibers ohne zusätzliche Kosten für unsere Kunden zur Verfügung stellen können.

### **2.2 Visualisierung**

Die JUNG Facility-Pilot – Visualisierung kommt bei Projekten mit einer großen Anzahl von Arbeitsblättern (Prozessbildern) zum Einsatz. Um in diesen Fällen den Hauptspeicherbedarf für die Visualisierung zu begrenzen wird nun die Anzahl der offenen Arbeitsblätter beschränkt. Wenn durch das Öffnen eines Arbeitsblatts die maximale Anzahl von offenen Arbeitsblättern überschritten wird, dann wird das zeitlich am längsten nicht verwendete Arbeitsblatt geschlossen – und später bei Bedarf wieder geöffnet.

Die Liste der Bildschirmauflösungen für das Zielsystem wurde erweitert, insbesondere um Breitwand - Formate.

Im Rahmen der neuen Netzwerkfähigkeit des Systems können nun Visualisierungsplayer über TCP/IP-Verbindungen mit einem entfernten Prozessmodell verbunden werden. Eine typische Anwendung davon sind mehrere Touchpanels mit einer zentralen Station, die mit dem technischen Prozess verbunden ist.

Der Betrieb in einem Netzwerk als Client oder als Server ist nicht in allen Produktvarianten möglich.

## 2.3 Prozessmodelle

Bitte lesen Sie genau, welche Änderungen am Prozessmodell vorgenommen wurden. Das Verhalten des Prozessmodells von bereits vorhanden Projekten muss in Hinblick auf die Änderungen überprüft werden!

*Vorbemerkung:* Die Abarbeitung des Prozessmodells erfolgt in Zyklen. Jeder Zyklus hat folgende prinzipielle Struktur:

- Lesen aller Eingangsvariablen aller Funktionen des Prozessmodells (Logische Funktionen, Timer, ..., Skripte)
- Auswerten aller Funktionen
- Zurückschreiben der Änderungen

Als Voraussetzung für Auswertung einer Funktion müssen ihre Eingangsvariablen mit Werten belegt sein. Hat eine Prozessvariable noch keinen Wert, z. B. weil ihr Wert noch nicht aus dem technischen Prozess ausgelesen wurde, dann können auch Funktionen noch nicht ausgewertet werden, zu denen diese Prozessvariable eine Eingangsvariable ist.

### – Basic-Skripte

An den Basic-Skripten wurden verschiedene Änderungen vorgenommen:

- Bisher konnte in einem Basic-Skript des Prozessmodells eine Prozessvariable nicht gleichzeitig als Eingangs- und als Ausgangsvariable für das Skript verwendet werden.

Diese Einschränkung wurden in Version 2.0 aufgehoben. Dadurch ist es jetzt leichter Skripte zu schreiben, die von einer Zustandsvariablen abhängen und den Zustand selbst ändern.

Nach wie vor werden aber Eingangsvariablen und Ausgabevariablen unterschieden. Änderungen der Eingangsvariablen führen zur erneuten Auswertung des Skripts im nächsten Zyklus, Ausgabevariablen dagegen nicht.

- Der Wert einer Ausgangsvariablen wurde nur dann an den Prozess ausgegeben, wenn er durch das Skript verändert wurde. Jetzt werden die Werte immer an den Prozess ausgegeben. Dadurch können zum Beispiel leicht Skripte geschrieben werden, die sich wiederholende 'EIN'-Befehle erzeugen.
- Nach wie vor sind Änderungen an Prozessvariablen für andere Funktionen erst im nächsten Zyklus sichtbar. Geändert wurde aber die interne Behandlung der Prozessvariablen innerhalb eines Skripts. Wünschenswert ist, dass eine Prozessvariable innerhalb eines Skripts nach Möglichkeit wie eine normale Basic-Variable behandelt werden kann, die nachdem eine Zuweisung erfolgt ist in folgenden Anweisungen auch den neuen Wert liefert. Das war bislang nicht der Fall, weil die Zuweisung auch aus Sicht des Basic-Skripts erst im nächsten Zyklus wirksam wurde.

Daher wurde das Verfahren nach folgendem Prinzip geändert: An Stelle der eigentlichen Prozessvariablen arbeitet das Skript intern auf einer Kopie, die sich wie eine Basic-Variable verhält und bei der Wertänderungen unmittelbar wirksam werden, nach Ende des Skripts wird der Wert der Kopie in die Prozessvariable übernommen.

## **– Startverhalten**

Beim Start des Prozessmodells wird jede Prozessvariable gemäß ihrer Einstellungen initialisiert, entweder mit einem Startwert oder dem zuletzt gespeicherten Wert belegt oder mit 'undefiniert' markiert.

Eine Funktion kann ausgeführt werden, wenn alle Eingangsvariablen mit einem Wert belegt sind. Auch unter dieser Voraussetzung wurde beim Start eine Funktion trotzdem nicht ausgeführt, bis die Eingangsvariablen mit neuen Werten belegt waren. Dadurch sollten Auswirkungen vermieden werden, die auf Grund veralteter Informationen ausgelöst werden und deshalb falsch sind.

Die Praxis hat aber gezeigt, dass in vielen Fällen die unmittelbare Auswertung der Funktionen beim Start, auch mit Initialwerten oder gespeicherten Werten, zweckmäßig ist, zum Beispiel wenn berechnete Prozessvariablen zur manuellen Vorgabe von Sollwerten verwendet werden.

Daher wird jede Funktionen jetzt beim Start sofort ausgewertet falls die Eingangsvariablen mit Werten belegt sind, auch ohne Notwendigkeit, dass die Variablen neue Werte erhalten müssen. Mit der Option 'Startverhalten' kann zu einer Funktion jetzt aber festgelegt werden, ob das beim Start berechnete Funktionsergebnis ignoriert werden soll.

Ein Beispiel mit einem AND-Gatter mit zwei Eingängen macht die Unterschiede zwischen dem alten und dem neuen Verfahren deutlich:

Beim bisherigen Verfahren müssen erst beide Variablen neue Werte erhalten, bevor die logische Verknüpfung ausgewertet wird.

Beim neuen Verfahren wird die AND – Funktion direkt berechnet. Das Ignorieren des beim Start berechneten Funktionsergebnisses führt dabei nicht zu demselben Verhalten wie das bisherige Verfahren, denn für das Auslösen der AND – Funktion im nächsten Zyklus genügt die Änderung von nur einer Eingangsvariablen.

## **– Änderungen an Funktionen**

Die Funktionen 'Logikgatter', 'Statusobjekt', 'Weiterleitung', 'Mathematische Funktion', 'Wenn-Dann', 'Sequenz', 'Szene', 'Speicherszene' und 'Zustandswächter' sind mit der Option 'Startverhalten' ausgestattet.

Bei den Funktionen 'Logikgatter', 'Mathematische Funktion', 'Wenn-Dann', 'Statusobjekt' und 'Weiterleitung' wurden weitere Optionen zur Verwendung des Ausgangswerts ergänzt. Jetzt besteht auch die Möglichkeit den Ausgangswert zu Senden wenn sich der Zeitstempel einer Eingangsvariable ändert, unabhängig vom Wert der Eingangsvariablen.

## **2.4 OPC Editor**

Der OPC - Client der JUNG Facility-Pilot - Visualisierung wurde um die Möglichkeit erweitert, an Stelle des gesamten Datenbestands eines OPC-Servers selektiv Datenpunktgruppen in das JUNG Facility-Pilot - System aufzunehmen.

Diese Änderung wurde erforderlich weil sich gezeigt hat, dass manche OPC - Server einen sehr großen Datenbestand führen, von dem nur Teile für ein konkretes JUNG Facility-Pilot - Projekt benötigt werden.

## **2.5 Musiksteuerung**

Für die JUNG Facility-Pilot – Visualisierung ist eine neue Komponente für die Verteilung von digitalem Sound in Gebäuden verfügbar.

Dabei werden SqueezeBox – Geräte der Firma Slim Devices, Inc. (siehe [www.slimdevices.com](http://www.slimdevices.com)) angesteuert. SqueezeBox ist ein Netzwerk - Musikplayer.

Über ein IP-Netzwerk (drahtgebunden oder über Funk) werden Audio-Streams von einem Server, dem SlimServer, an die SqueezeBox übertragen, an die SqueezeBox können aktive Lautsprecher angeschlossen werden.

Die JUNG Facility-Pilot - Musiksteuerung implementiert eine Ankopplung an den SlimServer und somit die Steuerung der SqueezeBox – Geräte als Prozessanschluss, ähnlich dem EIB- oder OPC- Editor.

.. KURZE BESCHREIBUNG DER DATENTYPEN

.. EIN PAAR ZEILEN, WAS MAN DAMIT MACHEN KANN (Szenen...)

Die Musiksteuerung ist nicht in allen Produktvarianten enthalten.

## **3 Schwerpunkte**

### **3.1 Einfachheit und Komfort**

Der erste Schwerpunkt des JUNG Facility-Pilot- Systems liegt auf der einfachen und komfortablen Bedienung und Anwendung in allen Projektphasen: Planung, Inbetriebnahme und Wartung.

Das System besteht aus verschiedenen Funktionsmodulen, die jeweils übersichtlich eine Aufgabenstellung behandeln: Konfigurationsprogramme für die Prozessanschlüsse, Editoren für die Logik, die das System realisieren soll, Editor und Player für Prozessbilder, Kalenderprogramm, Systemsteuerung.

Alle Applikationen haben eine einheitliche Bedienoberfläche, die selbstverständlich nach modernen WINDOWS- Standards gestaltet ist. Wenn man sich mit einem Funktionsmodul vertraut gemacht hat, erkennt man die Bedienelemente auch in den anderen Funktionsmodulen gleich wieder. Die Konsequenz in den Bedienoberflächen ist daher eine wichtige Grundlage für die einfache Handhabung.

Ein weiteres Mittel sind die kontextsensitiven Werkzeuge, durch die der Benutzer jederzeit und in jedem Bearbeitungsschritt die wichtigen Informationen und Funktionen zur Verfügung hat.

Im Vergleich zu modernen Büroprogrammen, die einen vergleichsweise einfachen Gegenstand haben, erfordert die Behandlung von technischen Prozessen nach unserer Auffassung Bedienungsformen, die über die Standards von Büroprogrammen hinaus gehen. Das ist unser Ziel.

### **3.2 Flexibilität für wechselnde Anforderungen**

Der zweite Schwerpunkt des JUNG Facility-Pilot- Systems liegt auf der Flexibilität.

Die Anforderungen an eine Prozessvisualisierung unterscheiden sich oft. Das liegt nicht nur an unterschiedlichen Gegebenheiten im technischen Prozess, sondern insbesondere auch an unterschiedlichen Anwendungen auf Basis des technischen Prozesses.

Mit den Funktionsmodulen des JUNG Facility-Pilot- Systems können viele Funktionen einfach konfiguriert werden. Die möglichen Anwendungsszenarien sind aber nicht alle vorhersehbar, so dass Bedarf nach zusätzlichen Funktionsmodulen entstehen kann.

Aus diesem Grund ist das System von vornherein als offenes System aufgebaut, in das bei Bedarf auch kundenspezifische Funktionsmodule integriert werden können.

Spezielle Anforderungen können ggf. durch eine vollständig integrierte Plattform für die Programmierung mit Basic, kompatibel zu **Visual Basic for Applications** (VBA, siehe zum Beispiel Microsoft Office) gelöst werden.

Für häufige Aufgabenstellungen halten wir die Programmierung mit Scripts allerdings nicht für die beste Lösung. Dazu wollen wir das System auch in Zukunft um Funktionsmodule ergänzen, mit denen die Lösung einfach konfiguriert wird. Lassen Sie uns wissen, was Sie für Ihre konkrete Anwendung benötigen!

## 4 Technische Voraussetzungen

### 4.1 Hardware

<b>Prozessor</b>	Pentium IV oder gleichwertig Mindestens 1,2 GHz.
<b>Hauptspeicher</b>	256 MB
<b>Freie Festplattenkapazität</b> (hängt wesentlich von den Archiven ab, die konfiguriert werden)	40 GB
<b>Bildschirmauflösung</b>	1024 x 768 Pixel
<b>Farben</b>	Farbtiefe mindestens 16 Bit pro Pixel.
<b>Schnittstelle</b>	Serielle oder USB Schnittstelle für die Verwendung mit FALCON

### 4.2 Betriebssysteme

WINDOWS XP, alle Versionen	JA
----------------------------	----

### 4.3 Prozessanschluss

Für den European Installation Bus (EIB) wird der FALCON- Treiber der EIBA (EIB Association) verwendet.

Optional ist das JUNG Facility-Pilot- System mit einem OPC (OLE for Process Control)- Client ausgestattet, so dass an Stelle von EIB oder zusätzlich dazu auch OPC- Server für den Prozessanschluss verwendet werden können, die für eine Vielzahl von Automatisierungssystemen erhältlich sind.

### 4.4 Import von Daten aus der ETS

ETS 2 Version 1.3	JA, Verwendung von ETS, "OPC- Export"
ETS 2 Version 1.1, 1.2 und 1.3	JA, in Datei ausgegebene Druckexports werden eingelesen.
Frühere ETS - Versionen	NEIN

### 4.5 Microsoft Internet Explorer, Version 6; DirectX, Version 9.0c

Sind erforderlich, können von der Setup- CD installiert bzw. aktualisiert werden.



## 5 Logischer Aufbau des Gesamtsystems

**Im JUNG Facility-Pilot- System werden die Applikationen nach drei Ebenen gegliedert:**

### ■ Ebene der Prozessanschlüsse:

Bei den Prozessanschlüssen geht es darum, den Anschluss an EIB oder ein anderes Automatisierungssystem zu konfigurieren. Für diese Aufgaben sind Kenntnisse der Prozessanschlüsse erforderlich. Ergebnis ist die funktionsfähige Kommunikation mit den vorhandenen technischen Prozessen.

### ■ Ebene der logischen Modelle:

Das logische Prozessmodell baut auf den Prozessanschlüssen auf. Dabei kann auch mehr als ein Prozessanschluss in einem Prozessmodell verwendet werden. Die Kommunikation mit den angeschlossenen technischen Prozessen ist hier nicht mehr die Kernfrage, sondern es geht darum, festzulegen, wie die Anlage insgesamt funktionieren soll.

Die Daten aus den Prozessanschlüssen können ergänzt werden um zusätzliche automatische Funktionen, die Festlegung von Archiven, der Festlegung von Störungsbedingungen und beispielsweise auch die Weiterleitung von Störungsmeldungen per E-Mail.

### ■ Anwendungsebene:

Die Anwendungsebene baut auf dem Prozessmodell auf und definiert Ansichten des Systems, durch die der Benutzer einen Überblick über die Anlage erhält und steuernd eingreifen kann. Das sind Prozessbilder und andere Auswertungen. Dieser Ebene sind auch die Kalenderprogramme zugeordnet.

Die Rechte eines Benutzers für die Einflussnahme auf die einzelnen Anwendungen werden durch die Benutzerverwaltung festgelegt.



## **5.1 EIB- und OPC- Prozessanschluss**

Der EIB Editor realisiert den EIB- Prozessanschluss für das System. Dazu können Daten aus der ETS (EIBA Tool für EIB- Projektierung) in den EIB Editor importiert werden. Darüber hinaus hat der EIB Editor Funktionen zur Inbetriebnahme und Protokollierung.

Alternativ oder zusätzlich kann der technische Prozess über OPC (OLE for Process Control) angekoppelt werden. Dazu ist eine Vielzahl von OPC- Servern erhältlich, die den eigentlichen Prozessanschluss realisieren und für den OPC- Client des JUNG Facility-Pilot-Systems bereitstellen.

## **5.2 Prozessmodelle**

Der Prozessmodell- Editor konfiguriert ein Prozessmodell und importiert dazu Daten aus dem EIB- bzw. OPC- Editor. Dabei können in ein Prozessmodell zugleich unterschiedliche technische Prozesse integriert werden und zusätzliche Datenpunkte und Logikfunktionen konfiguriert werden.

Das Prozessmodell stellt eine logische Sichtweise auf das Projekt zur Verfügung, unabhängig von den Eigenschaften der einzelnen Prozessanschlüsse. Im Prozessmodell können ‚virtuelle Geräte‘ erzeugt werden, Archive definiert, E-Mail Benachrichtigungen konfiguriert und insgesamt also festgelegt werden, wie das technische System auf Basis der Prozessanschlüsse funktionieren soll.

Für den Ablauf des Projekts wird eine Applikation zur Ablaufsteuerung des Prozessmodells gestartet, die keine Bedienoberfläche hat. Der Prozessmodell- Editor wird dazu nicht benötigt.

## **5.3 Kalender**

Das Kalenderprogramm erlaubt das komfortable Erstellen und Konfigurieren automatischer Zeitschaltfunktionen in Verbindung mit einem Kalender.

## **5.4 Visualisierung**

Die Visualisierung durch Prozessbilder wird durch den Visualisierungs- Editor und den Visualisierungs- Player realisiert. Der Editor enthält zahlreiche Funktionen zur Gestaltung der Prozessbilder, der Player zeigt die Prozessbilder lediglich an und erfordert nur minimale Kenntnisse. Zum Ablauf des Projekts ist der Editor nicht erforderlich.

## **5.5 Systemsteuerung**

Die Systemsteuerung integriert die verschiedenen Funktionsmodule, Dokumentationen, die Projektverwaltung, projektübergreifende Einstellungen, und ist ein guter Ausgangspunkt, um die verschiedenen Funktionsmodule aufzurufen.

## **6 Betriebsarten**

Das JUNG Facility-Pilot- Paket kann in einer von drei Betriebsarten laufen und lässt, je nach Produktversion, den Betrieb mit einer bestimmten Anzahl von EIB- Datenpunkten zu.

Das Produkt muss für einen Kunden und einen bestimmten Rechner durch einen Aktivierungs- Code freigeschaltet werden.

### **6.1 Als DEMO- Version:**

Der Prozessanschluss funktioniert nur eine begrenzte Zeit, dann wird die Verbindung zum Prozess getrennt.

Die DEMO- Version ist grundsätzlich begrenzt auf eine Anzahl von Datenpunkten, einen bestimmten Zeitraum, und eine Anzahl von Programmstarts innerhalb dieses Zeitraums.

### **6.2 Als aktivierte Produktversion:**

Als aktivierte Produktversion ist das JUNG Facility-Pilot- System nicht zeitlich oder durch die Anzahl der Programmstarts begrenzt. Die Anzahl der Datenpunkte ergibt sich aus der gekauften Produktversion.

### **6.3 Als nicht- aktivierte Produktversion:**

Als nicht- aktivierte Produktversion läuft das JUNG Facility-Pilot- System wie die aktivierte Produktversion, aber begrenzt auf einen Zeitraum und eine Anzahl von Programmstarts.

## **7 Kopierschutz und Aktivierung**

Das JUNG Facility-Pilot- System verfügt über einen Kopierschutz, der an die Hardware des Rechners gebunden ist, auf dem das System installiert wurde. Die Software wird auf dem Rechner durch einen Aktivierungs- Code aktiviert.

Den Aktivierungs- Code liefert der Händler passend zu einem System- Code, der zuvor aus der Hardware und der Produktversion berechnet wurde. Der Kopierschutz hängt nicht ab von Speicherausbau, Netzwerkkarte, Grafikkarte, Soundkarte oder anderen Zusatzgeräten.

Jede Kombination aus System- Code und Aktivierungs- Code kann nur einmal verwendet werden.

Wenn zu einem aktivierten Produkt die Hardware des Rechners verändert wird, dann ist eine erneute Aktivierung erforderlich. In diesem Fall geht das System in eine zeitbegrenzte Betriebsart über, in der alle anderen Produkteigenschaften weiter vorhanden sind.

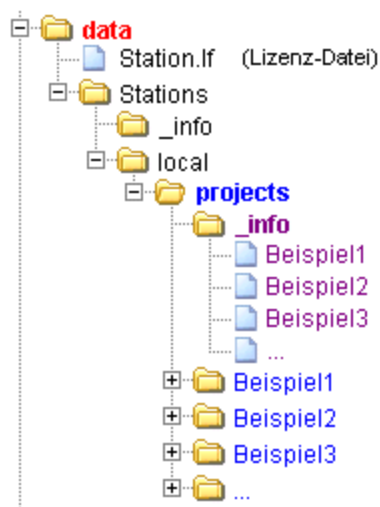
## 8 Ablage von Projekten

Diese Beschreibung der Dateiorganisation dient zur allgemeinen Information. Kenntnisse der Dateiorganisation sind für den Betrieb des JUNG Facility-Pilot- Systems nicht erforderlich.

Für das Übertragen von Projekten zwischen verschiedenen Rechnern raten wir davon ab, Projekte direkt als Dateien zu kopieren. Zu diesem Zweck gibt es die Import/Export-Funktionen der JUNG Facility-Pilot Systemsteuerung.

Bei der Installation wird ein Verzeichnis für sämtliche Projektdaten des Systems festgelegt. Das Verzeichnis hat den Namen **DATA**.

**Ausgehend von DATA ergibt sich folgende Verzeichnisstruktur:**



Innerhalb **PROJECTS** hat jedes Projekt ein Unterverzeichnis mit dem Projektnamen.

Außerhalb enthält das Unterverzeichnis **\_INFO** zu jedem Projekt eine Datei mit dem Projektnamen.

## 9 Datensicherung und Notbetrieb

Als Datensicherung sollte das Verzeichnis **DATA** in regelmäßigen Zeitabständen vollständig gesichert werden.

Wenn dann Ihr Rechner nicht mehr funktionsfähig ist, können Sie die Software neu auf einen Ersatzrechner installieren und anschließend das gesicherte **DATA**- Verzeichnis in das **DATA**- Verzeichnis des Ersatzrechners kopieren.

## 10 Anschrift

### Ansprechpartner

#### **ALBRECHT JUNG GMBH & CO. KG**

Volmestraße 1  
58579 Schalksmühle

Telefon +49 (0) 23 55/80 60

Telefax +49 (0) 23 55/80 61 89



[mail.info@jung.de](mailto:mail.info@jung.de)

### Copyright

Copyright ©2006 ESF Software GmbH  
Alle Rechte vorbehalten

### Warenzeichen

EIB® ist ein eingetragenes Warenzeichen der EIB association (EIBA).

LON® ist ein eingetragenes Warenzeichen der Echelon Corporation registriert in Amerika und anderen Ländern.

OPC® ist ein eingetragenes Warenzeichen der OPC Foundation.

Sax Basic Engine ist ein eingetragenes Warenzeichen der Sax Software Corporation.

Adobe Acrobat® ist ein eingetragenes Warenzeichen der Adobe Systems Incorporated.

Microsoft®, ActiveX®, DirectX®, Windows®, Windows NT®, Excel®, Visual Basic® sind eingetragene Warenzeichen der Microsoft Corporation.

Alle Handelsnamen, Firmennamen und Produktnamen sind Warenzeichen oder eingetragene Warenzeichen ihrer jeweiligen Besitzer.