

cyProvider®

INSTALLATIONS- UND ADMINISTRATIONS- LEITFADEN

Stand 23.09.2009

© cyFlex GmbH
Maastrichter Straße 24 • DE-41464 Neuss
Telefon: +49 2131 980501 • Telefax: +49 2131 980502
E-Mail: info@cyflex.de

Inhaltsverzeichnis

1. Einleitung	3
1.1. Was ist cySystem?	3
1.2. Woraus besteht cySystem?	3
1.2.1. cyAgent	3
1.2.2. cyProvider	4
2. Installation cyProvider	5
2.1. cyProvider-Installer	5
2.1.1. cyProvider Verzeichnisstruktur	5
2.1.2. Programme, Libraries, ini-Files	7
2.1.3. Installation der Beispielanwendung Digitaler Atlas	7
2.1.4. Installation von cyProvider als Windows-Dienst	8
2.2. Überprüfen der Installation	9
3. Administration von cyProvider	10
3.1. conf.ini	10
3.2. cyProvider.ini	11
3.2.1. Konfigurationstool / Parameter	11
3.2.2. Windows-Logon / Datenbank-Logon / UserImage	11
3.2.3. Definition der benötigten Provider	13
3.3. Einsatz des SSL-Protokolls	14
3.4. Installation von Anwendungen	14
3.4.1. Sources-Verzeichnis	15
3.4.2. Images-Verzeichnis	17
3.5. Aufruf von Anwendungen mit dem cyAgent	18
3.6. Übersicht der zu administrierenden Dateien	18

1. Einleitung

Zunächst folgt ein kurzer Überblick zur Einordnung von cyProvider als eine der beiden Hauptkomponenten von cySystem.

1.1. Was ist cySystem?

cySystem ist zugleich Client/Server-Anwendungsgenerator zur Entwicklung sowie WAN-Infrastruktur zum Betrieb von internetfähigen Datenbankanwendungen. Im Vergleich zu anderen Web-basierenden Tools erstellt cySystem echte Windows-Anwendungen, die ohne Web-Browser auf dem Client-PC laufen.

1.2. Woraus besteht cySystem?

cySystem besteht aus dem cyAgent als Front- und dem cyProvider als Backend. Hinzu kommt noch die cyToolbox, eine integrierte Entwicklungsumgebung für die Erstellung von cySystem -Anwendungen.

1.2.1. cyAgent

Der cyAgent ist ein Proxy-fähiger Applikations-Browser, der, im Unterschied zu einem Web-Browser, echte Windows-Anwendungen präsentiert. Er arbeitet lokal als Client auf dem PC der Anwender. Zu seinen Aufgaben gehören:

- Kommunikation mit dem cyProvider

Der cyAgent verwendet das SOAP-Protokoll (mit oder ohne SSL) zur Kommunikation mit dem Server. Eine leistungsfähige Compress Engine sorgt für kurze Übertragungszeiten. Bereits mit einfacher DSL-1000 Übertragungsrate zum Internet ist der Unterschied zu einer lokalen Verbindung im LAN nicht mehr spürbar.

- Interpretation der Definitionsdateien

cySystem-Anwendungen werden nicht programmiert, sondern definiert. Die Eigenschaften der Anwendungen werden in Definitionsdateien (Textdateien) beschrieben, die keinerlei ausführbaren Code beinhalten und leicht erlernbar sind. Dieses Verfahren hat viele Vorteile. Das Risiko durch einen Hacker-Angriff Daten zu verlieren oder Dritten zugänglich zu machen ist erheblich geringer als bei Anwendungen, die Java oder ActiveX einsetzen. Die einfache Definitionssyntax erlaubt es, auch ohne Programmiererfahrung eine Anwendung zu erstellen. Der größte Vorteil aber sind die extrem kurzen Entwicklungszeiten. Nicht mehr Mann-Monate sind die Größenordnung für komplexe Datenbankanwendungen, sondern lediglich Tage.

Der aktuelle cyAgent bestehend aus der cyAgent.exe und den Libraries libeay32.dll und libssl32.dll steht im Internet auf der Seite www.cyflex.de als Download zur Verfügung. Er ist in ein beliebiges lokales Verzeichnis zu kopieren und durch Doppelklick der exe zu starten. Alternative Aufrufmethoden, insbesondere der direkte Aufruf von einzelnen cySystem-Anwendungen sind im Kapitel 3.5 beschrieben.

1.2.2. cyProvider

Der cyProvider ist der Applikations-Server, der als Windows-Dienst auf einem Zentralsystem installiert wird. Zu seinen Aufgaben gehören:

- Benutzerverwaltung

Benutzer können sich an einer Windows-Domäne anmelden oder einfach nur mit einer Datenbank anmeldung authentifizieren.

- Informationsverwaltung

Der cyProvider unterstützt alle strukturierten Datenquellen. Hierzu gehören beispielsweise Datenbanken, Tabellenkalkulationen, Textdaten und auch Anwendungen, die in der Lage sind, Daten strukturiert anzubieten (z.B. SAP-RFC).

- Anwendungsverwaltung

Die Definitionsdateien liegen als Quelltext oder wahlweise auch als Kompilat (Image) zentral auf dem Server. Die Anwendung wird erst auf Anforderung eines berechtigten Benutzers auf dessen cyAgent herunter geladen und verbleibt für die Dauer der Anwendung im Hauptspeicher auf dem Client.

Im Folgenden werden die für die Installation und Administration des cyProvider wesentlichen Punkte näher beschrieben.

2. Installation cyProvider

Zur Installation von cyProvider steht der cyProvider-Installer in Form einer Setup-Routine zur Verfügung.

2.1. cyProvider-Installer

Im Full Installation Mode werden vom cyProvider-Installer folgende Aktivitäten durchgeführt:

- Anlegen der benötigten Ordner im Installationsverzeichnis
- Kopieren der Programm- und ini-Files
- Installation der Beispielanwendung Digitaler Atlas
- Installation und Starten des Windows-Dienstes cyProvider

Im Custom Installation Mode kann die Beispielanwendung Digitaler Atlas separat installiert werden. Zusätzlich besteht die Möglichkeit in einem bereits installierten System nur das Programm und die Libraries zu aktualisieren.

2.1.1. cyProvider Verzeichnisstruktur

cyProvider benötigt eine fest vorgegebene Verzeichnisstruktur (siehe Abb. 1). Das root-Verzeichnis, in dem sich die cyProvider.exe, die Libraries und die beiden Dateien *conf.ini* und *cyProvider.ini* befinden, enthält folgende Unterverzeichnisse:

- Cert

Cert enthält Server- und Root-Zertifikat und den private Key zum Aufbau einer SSL-Verbindung. Mit der Installation wurden Zertifikate und Schlüssel zu Testzwecken im Verzeichnis abgelegt.

- Images

In diesem Verzeichnis befinden sich die mit dem Tool *cylmager* vorkompilierten Definitionen der Applikationen, sowie die Definitionen für den *cyAgent-Browser* (siehe Kapitel 3.4.1.2). Ein Image für die Beispielanwendung Digitaler Atlas wird standardmäßig mit der Installation zur Verfügung gestellt. Der Parameter *Uselimage* in der *cyProvider.ini* steuert, ob Anwendungen aus dem Images-Verzeichnis oder aus dem Sources-Verzeichnis zum anfordernden *cyAgent* geschickt werden.

- Sources

Das Verzeichnis Sources enthält die Definitionen der Applikationen in lesbaren Textdateien in jeweils eigenen Unterverzeichnissen. Für die Beispielanwendung Digitaler Atlas wurde das Unterverzeichnis DigiAtlas angelegt. Dieses Verzeichnis enthält wiederum Unterverzeichnisse für die rollenunabhängigen (Def-Files) und rollenabhängigen (Roles) Definitionen (Def-Files), sowie ein Verzeichnis für anwendungsspezifischen Sprachdefinitionen (Language).

Das Unterverzeichnis Browser enthält die Definitionen für den cyAgent-Browser (siehe Kapitel 3.4.1.2). Neben einer Kurzbeschreibung im rtf-Format und einem Anwendungslogo für die Präsentation im cyAgent-Browser wird für jede verfügbare Anwendung eine ini-Datei für den download zum cyAgent bereitgehalten (z.B. *DigiAtlas.ini*). Diese ini-Datei beschreibt für die gewünschte Anwendung das Anmeldeverfahren (mit oder ohne Autologon) und bestimmt den Logon-Provider inkl. seiner Adresse (URL).

Im Unterverzeichnis Globals befinden sich allgemein gültige Einträge. Aktuell sind das die anwendungsübergreifenden Sprachdefinitionen (Fehlertexte, Systemmeldungen, Bezeichnungen der Buttons, etc.), die im Ordner Language je Sprache in Dateien mit dem Namen `<countrycode>_Globals.cdd` abgelegt sind.

- Template

Dieses Verzeichnis enthält ein Template zum Erstellen zusätzlicher Anwendungen. Näheres dazu ist dem [cySystem – How-to-Guide](#) zu entnehmen.

Die ebenfalls vom cyProvider-Installer angelegten Unterverzeichnisse Data und Logs können nach Änderung der entsprechenden Parameter in der *cyProvider.ini* bzw. in der *conf.ini* an beliebige Orte verschoben werden. Data enthält die Datenbank für die Beispielanwendung und Logs die Ticket- und Trace-Logfiles.

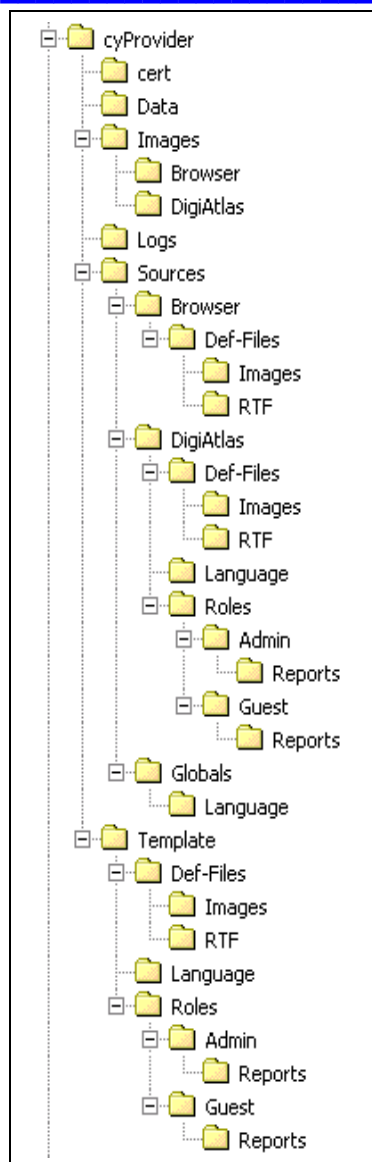


Abb. 1

2.1.2. Programme, Libraries, ini-Files

Neben dem Hauptprogramm cyProvider.exe wird für das Datenhandling die Library midas.dll benötigt. Die beiden Libraries libeay32.dll und ssleay32.dll sind für SSL-Verbindungen zum cyAgent notwendig. Die Konfiguration von cyProvider erfolgt durch Änderungen der Parameter in den beiden ini-Dateien *conf.ini* und *cyProvider.ini* (siehe Kapitel 3).

2.1.3. Installation der Beispielanwendung Digitaler Atlas

Um Arbeitsweise und Funktionalität von cyProvider unmittelbar testen zu können, sollte die Beispielanwendung Digitaler Atlas mitinstalliert werden. Diese Anwendung demonstriert die Möglichkeiten von cySystem am Beispiel einer n:m Länder-Städte-Beziehung. Als Datenquelle dient die MS Access Datenbank DemoDB im Verzeichnis Data. Die Authentifizierung erfolgt gegen die Tabelle USR der Datenbank DemoDB. Die User Admin und Demo mit jeweils gleichlautendem Passwort sind den Rollen ADMIN bzw. GUEST zugeordnet. Aus Berechtigungssicht unterscheiden sich die beiden Rollen darin, dass dem User Admin zusätzlich der Menüpunkt Benutzerpflege zugeordnet ist.

Voraussetzung für den Datenbankzugriff ist die installierte Microsoft Jet Engine. Die aktuelle Jet Engine kann über das Jet Engine Servicepack 8 installiert werden.

2.1.4. Installation von cyProvider als Windows-Dienst


Sollte die Installation von cyProvider als Windows-Dienst fehlgeschlagen sein, kann er auch manuell durch Eingabe von <root-Verzeichnis>cyProvider.exe /install in der Befehlszeile installiert bzw. durch /uninstall deinstalliert werden. Anschließend sollte cyProvider im Dienstemanager erscheinen. Unter Eigenschaften/Anmelden des Dienstes (Abb. 2) müssen Sie den "Datenaustausch zwischen Desktop und Dienst zulassen", damit Sie das Konfigurationstool aufrufen und cyProvider nach Ablauf der Trial-Phase registrieren können. Nach dem erneuten Start des Dienstes erscheint dann das Icon  des Konfigurationstools von cyProvider in der Taskbar des Rechners.



Abb. 2

2.2. Überprüfen der Installation

Anhand der folgenden Aktivitäten kann die Installation von cyProvider überprüft werden:

- Im Dienstemanager (Start/ Einstellungen/ Systemsteuerung/ Verwaltung/ Dienste) überprüfen, ob der Dienst gestartet wurde.
- In der Datei *cyProviderTrace.log* im Verzeichnis Logs kann überprüft werden, ob und mit welchen Einstellungen der SSL- bzw. der non SSL-Server gestartet wurde.
- Aufruf der Beispielanwendung Digitaler Atlas mit dem cyAgent.
- Überprüfen der Einstellungen in der *conf.ini* und *cyProvider.ini* bezüglich cyProvider-Problemen bzw. in der *DigiAtlas.ini* und der *cyGuiApl.ini* bei Problemen mit der Beispielanwendung.

Die häufigste Fehlerursache dafür, dass sich cyProvider nicht als Dienst starten lässt, liegt an der Nutzung der Ports 80 bzw. 443 durch andere Programme. In diesem Fall müssen alternative Ports in der *conf.ini* eingetragen werden.

Wichtig: Werden vom Standard abweichende, alternative Ports benutzt, müssen diese Ports im cyAgent-Browser in der cyProvider-URL mit vorangestelltem Doppelpunkt angehängt werden.

3. Administration von cyProvider

Die Administration von cyProvider erfolgt in erster Linie über die Pflege der beiden ini-Dateien *conf.ini* und *cyProvider.ini*. Darüber hinaus sind noch Aktivitäten für die Nutzung des SSL-Protokolls und natürlich für die Installation weiterer Anwendungen erforderlich.

3.1. conf.ini

Die *conf.ini* enthält die für den Serverbetrieb grundsätzlichen Parameter:

- rootpath=[Installationsverzeichnis von cyProvider]
- logall=n [y=maximale Log-Ausgabe]
- tracelogfile=[vollständiger Pfad\Filename des Tracelogs]
- tracelog=y [default=n]; aktiviert das Schreiben eines Tracelogfiles
- ticketlogfile=[vollständiger Pfad\Filename des Ticketlogs]
- ticketlog=y [default=n]; aktiviert das Schreiben eines Ticketlogfiles
- MaxTicketAge=3600 [Gültigkeitsdauer eines Tickets in Sekunden]
- portnonssl=80 [Port für den non SSL Server]
- portssl=443 [Port für den SSL Server]
- certpassword=mypwd [Zertifikatspasswort für das Serverzertifikat im Verzeichnis Cert, mypwd ist das Passwort für das zu Testzwecken erstellte Zertifikat]
- Compression=True [Datenkomprimierung aktiv]
- ProviderName=cyProvider [Dienstname]
- DisplayName=cyProvider [Anzeigename]
- license=[Lizenzschlüssel für Multi-User Betrieb]

Nach einer erfolgreichen Authentifizierung stellt der cyProvider ein Ticket für den Benutzer aus und schickt dieses zum cyAgent zurück, der mit diesem Ticket berechtigt wird genau eine weitere Anfrage an den cyProvider zu schicken. Nach jeder Anfrage erhält der cyAgent ein neues Ticket vom cyProvider, dessen Gültigkeitsdauer durch den Parameter


MaxTicketAge beschränkt wird. Ist die Gültigkeitsdauer des Tickets abgelaufen, muss der Anwender die Anwendung neu starten und sich wieder authentifizieren.

Wichtig: Nach jeder Parameteränderung muss der cyProvider-Dienst neu gestartet werden, damit die Änderung wirksam wird!

3.2. cyProvider.ini

Die Parameter der *cyProvider.ini* bestimmen das Authentifizierungsverfahren, den Einsatz von Images und stellen Provider bereit, die die Verbindungen zu den Datenbanken liefern.

3.2.1. Konfigurationstool / Parameter

Die Sektion [Locale ID: 1031] enthält die Beschriftungen des Konfigurationstools, mit dessen Hilfe die benötigten Provider (siehe Kapitel 3.2.3) ebenfalls bearbeitet werden können. Der obige Name der Sektion bezieht sich auf einen deutschsprachigen Desktop. Ist am Desktop zum Beispiel Englisch (United Kingdom) als Sprache eingestellt, muss eine Sektion mit dem Namen [Locale ID:2057] mit den Beschriftungen vorhanden sein. Das Konfigurationstool wird über das Icon  in der Taskbar aufgerufen (siehe Kapitel 2.1.4).

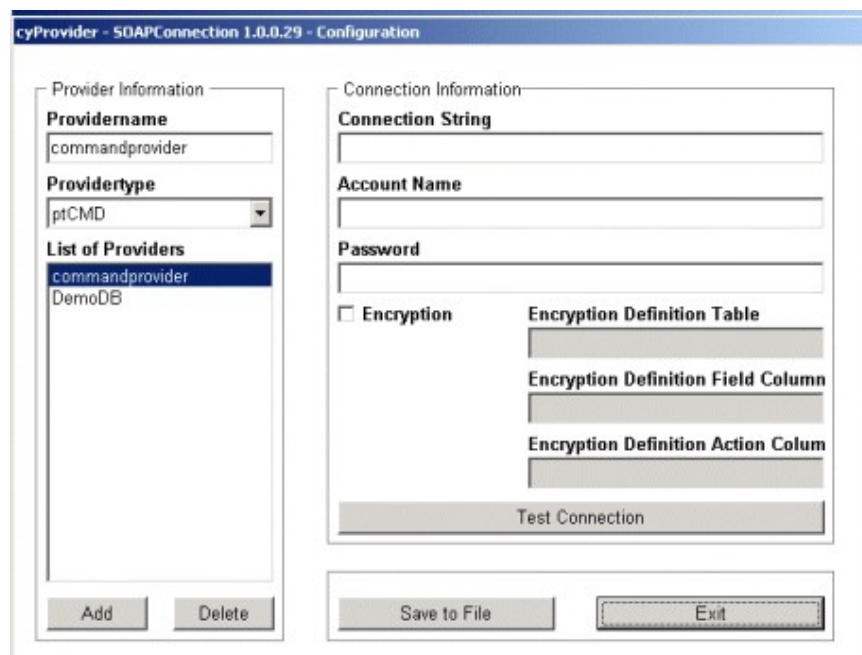


Abb. 3

3.2.2. Windows-Logon / Datenbank-Logon / Uselimage

Die folgenden Parameter können nur direkt in der *cyProvider.ini* gepflegt werden:

- UselImage=n [y => es werden die vorkompilierten Applikationen aus dem Image-Verzeichnis anstelle der Definitionsdateien aus dem Source-Verzeichnis zum cyAgent gesendet.]
- LogonProvider=Windows [Windows Logon]

Voraussetzung für die Authentifizierung gegenüber Windows ist, dass für jede Rolle der Anwendung eine Benutzergruppe <Applikationsname.Rollenname> angelegt wurde, die wiederum die berechtigten Benutzer enthält.

Die Rolle GUEST ist für den anonymen Gast reserviert, der einen bestimmten, öffentlichen Teil der Applikation auch ohne Anmeldung nutzen darf (siehe Parameter AutoLogon=True in der ini-Datei der Anwendung).

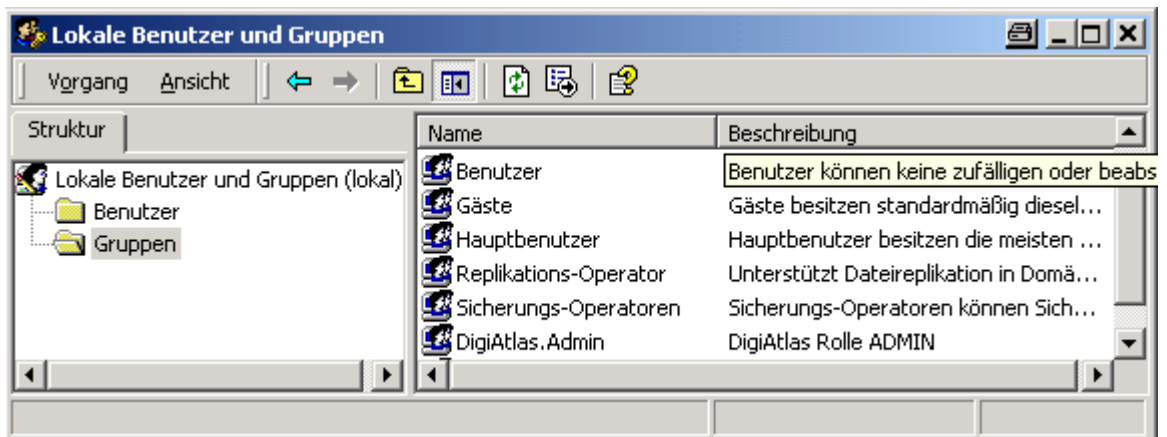


Abb. 4

- LogonProvider=DemoDB [Datenbank-Provider für den Datenbank-Logon]

Alternativ kann die Authentifizierung auch gegen eine Datenbank laufen, die über einen Provider (im Beispiel DemoDB) verbunden ist und die Accountdaten enthält. Die Zuordnung zu den Accountdaten erfolgt über die folgenden Parameter (Beispiel Tabelle USR der Datenbank DemoDB):

- LoginSQL=SELECT * FROM USR WHERE USR.USR_LOGON = :UN AND USR.USR_APPLICATION = :APP

SQL-Anweisung zur Ermittlung der Accountdaten für den eingegebenen Usernamen, der in der Variable :UN gespeichert wurde und der aufgerufenen Anwendung (:APP). Bei case-sensitiver Datenbank muß der Username in Großbuchstaben angelegt werden.

- UserField=USR_LOGON ;welches Feld enthält den Accountnamen

-
- PasswordField=USR_PASSWORD ;welches Feld das Passwort
 - RoleField=USR_ROLE ;welches Feld den Rollennamen
 - AppField=USR_APPLICATION ;welches Feld den Anwendungsnamen

3.2.3. Definition der benötigten Provider

Die von den Anwendungen, die der cyProvider bereitstellt, benötigten Datenbank-Provider können sowohl direkt in der *cyProvider.ini* unter der Sektion MDC als auch über das Konfigurationstool eingepflegt werden. Letzteres speichert die Einstellungen ebenfalls in der *cyProvider.ini*, bietet aber zusätzlich die Möglichkeit eines Verbindungstests zur Datenbank.

Zur Zeit werden drei verschiedene Providertypen unterstützt:

- CMD

Der Commandprovider vom Typ CMD ist standardmäßig definiert. Er dient zur Kommunikation zwischen dem cyProvider und dem cyAgent.

- ADO (Microsoft ActiveX Data Objects)

Mit dem ADO-Provider ist der Zugriff auf eine Vielzahl von Datenbanksystemen (Access, SQL Server, Oracle, ...) oder Anwendungen wie Excel, Exchange bis hin zu reinen Textdateien möglich.

- SAP

Der SAP-Provider wurde von cyFlex entwickelt. Im Unterschied zum ADO-Provider greift er nicht direkt auf die Datenbank des SAP-Systems zu, sondern ruft über die SAP RFC-Schnittstelle spezielle Funktionsbausteine auf, die die Daten bereitstellen bzw. Änderungen entgegen nehmen. Damit verbleibt die Kontrolle über die Daten beim SAP-System.

Das folgende Beispiel zeigt die notwendigen Definitionen für die beiden Provider DemoDB und ProviderSAP:

```
[MDC]
```

```
Providers=DemoDB|ProviderSAP
```

```
[DemoDB.PreInit]
```

```
;ALTER SESSION SET NLS_NUMERIC_CHARACTERS = ",."
```

```
[MDC.DemoDB]
```

```
ProviderType=ptADO
ConnectionString=Provider=Microsoft.Jet.OLEDB.4.0;
    Data Source=C:\Programme\cyProvider\Data\DemoDB.mdb;
    Persist Security Info=False
DatabaseAccount=
DatabasePassword=
```

[MDC.ProviderSap]

```
ProviderType=ptSAP
ConnectionString=Server=myHostname;Client=000;
    User=%DatabaseAccount%;Password=%DatabasePassword%;
    System=MBS;SysNr=00;Lang=DE;LCID=1031
DatabaseAccount=aroiPHAYTEo=
DatabasePassword=DERofDh=
```

Für jeden, in der Providerliste der MDC-Sektion aufgeführten Provider, existiert eine separate Untersektion, in der der ProviderType, der ConnectionString und gegebenenfalls der Datenbank-Account und das Passwort anzugeben sind.

Zusätzlich können in einer Prelnit-Sektion Initialisierungen veranlaßt werden, bevor der Provider aktiv ist. Wenn z.B. die Datenbank in einer anderen Sprachen betrieben wird als der Sprache des Betriebssystems unter dem der cyProvider läuft, können in der Prelnit-Sektion entsprechende Spracheinstellungen vorgenommen werden.

Wichtig: Nach jeder Parameteränderung muss der cyProvider-Dienst neu gestartet werden, damit die Änderung wirksam wird!

3.3. Einsatz des SSL-Protokolls

cyProvider unterstützt eine SSL-Verbindung zwischen cyAgent und cyProvider auf Basis von OpenSSL. Dazu müssen das Root-Zertifikat, das Server-Zertifikat und der private Schlüssel im PEM-Format im Verzeichnis Cert abgelegt werden. Mit der Installation werden Zertifikate bzw. ein privater Schlüssel zu Testzwecken hinterlegt. Diese sind vor dem produktiven Einsatz durch eigene Zertifikate/Schlüssel zu ersetzen.

3.4. Installation von Anwendungen

Wie bereits erwähnt, werden Anwendungen vom cyProvider in Form von Definitionsdateien beschrieben und verwaltet. Diese Definitionsdateien (Dateityp cdd) werden berechtigten Benutzern über das Netzwerk (LAN/WAN) zur Verfügung gestellt und auf Clientseite vom cyAgent interpretiert und als Anwendung präsentiert. Die Definitionsdateien liegen entweder in lesbarer Form im Verzeichnis Sources oder als Kompilat im Verzeichnis Images. Beide Verzeichnisse haben wiederum eine festvorgegebene Verzeichnisstruktur.

Neben den Definitionsdateien befinden sich je Anwendung noch zwei ini-Dateien in diesen Verzeichnissen. Sie enthalten die für den Start einer Anwendung benötigten Parameter. Die Aufteilung der Parameter in zwei ini-Dateien hängt mit dem Berechtigungskonzept zusammen.

Die erste ini-Datei mit der Bezeichnung *<Anwendungsname>.ini* enthält die Parameter für die gewünschte Authentifizierung und den Namen der zweiten ini-Datei (der Default-Name ist *cyGuiApl.ini*). Die *<Anwendungsname>.ini* liegt entweder im Source\Browser\Def-Files-Verzeichnis oder im Installationsverzeichnis des cyAgent (siehe Kapitel 3.5).

Erst nach erfolgreicher Anmeldung wird die zweite ini-Datei (*cyGuiApl.ini*), die im Def-Files-Verzeichnis der Anwendung liegt, zusammen mit den Definitionsdateien dem cyAgent zur Verfügung gestellt.

3.4.1. Sources-Verzeichnis

Neben den Unterverzeichnissen Browser und Globals enthält das Verzeichnis Sources für jede vom cyProvider unterstützte Anwendung ein separates Unterverzeichnis mit dem jeweiligen Anwendungsnamen.

3.4.1.1. Anwendungsverzeichnis

Ein Anwendungsverzeichnis gliedert sich wiederum in folgende Unterverzeichnisse:

- DEF-FILES

Dieses Verzeichnis enthält die rollenunabhängigen Definitionen der Anwendung.

- In der Datei *cyGuiApl.ini* (evtl. wurde ein anderer Name in der *<Anwendungsname>.ini* definiert) sind die für die Anwendung grundlegenden Parameter enthalten:
 - die von der Anwendung benötigten Provider und die zugehörigen Adressen (URLs),
 - die unterstützten Sprachen mit Sprach-ID=LCID, Datenbankfeldnamen-Erweiterung und Sprachbezeichnung (z.B. de=1031; 0; deutsch => Datenbankfelder mit deutschen Inhalten haben eine Bezeichnung der Form *Feldname_0*),
 - definierte Placeholder
- Die Beschreibungen der einzelnen Formulare bezüglich Größe, Struktur (horizontal oder vertikal unterteilt, mit oder ohne Tabreiter) und der enthaltenen Elemente (Links auf Selektions- oder Pflegeautomaten) liegen je Formular in einer separaten Datei vom Typ *<Formularname>.cdd*.

-
- Die Definition des Hauptmenüs ist in der Datei *<Anwendungsname>.cdd* enthalten.
 - Die Datei *_STYLES.cdd* enthält die vordefinierten Layout-Definitionen, die dem Anwender zur Auswahl zur Verfügung gestellt werden.
 - Neben den Zuordnungen von Styles zu bestimmten Formularelementen von Selektions- und Pflegeautomaten, sowie der Definition von Critical-Colors für die Kennzeichnung von Muss- oder nicht eingabebereiten Feldern, können in der Datei *_DESIGN.cdd* Style-Conditions abgelegt sein. Mit Hilfe von Style-Conditions kann das Layout von Feldern in Abhängigkeit von deren Inhalt dynamisch verändert werden.
 - In der Datei *_CONNECT.cdd* sind Definitionen für die Datenbank-Updates von Massenänderungen per Drag & Drop hinterlegt.
 - *_PROGRAMS.cdd* enthält die für den Aufruf von externen Programmen benötigten Definitionen.

Zusätzlich besitzt DEF-FILES noch die beiden Unterverzeichnisse Images und RTF. In Images sind Bitmaps für die Sprachen (*<Sprach-ID>.bmp*), die als Icon im Menüpunkt Sprachauswahl dienen und ein Bitmap Barglyph, das in der ersten Spalte der Untermenüs angezeigt wird, abgelegt. RTF enthält Dokumente im RTF-Format, die im Menüpunkt Info angezeigt werden sollen.

- Language

Das Language-Verzeichnis enthält pro definierter Sprache eine Datei *<Sprach-ID>.cdd*, in der alle anwendungsspezifischen sprachabhängigen Bezeichnungen (Überschriften, Feldnamen, Formulanamen, etc.) enthalten sind.

- Roles

Im Verzeichnis Roles befindet sich für jede definierte Rolle ein Unterverzeichnis mit den rollen-spezifischen Definitionen. Für jedes im Verzeichnis DEF-FILES zuvor definierte Formular, befindet sich im ROLES-Verzeichnis eine namensgleiche cdd-Datei, in der die zulässigen Inhalte mit Hilfe von SQL-Anweisungen definiert werden.

In der Datei *<Anwendungsname>.cdd* wird das rollenspezifische Detail-Menü definiert.

Wurden mit dem FastReport-Designer erstellte Reports der Rolle zugeordnet, so befinden sich die Report-Definitionen als XML-Datei im Unterverzeichnis Reports.

Für den anonymen Benutzer dient das Rollenverzeichnis mit dem reservierten Namen GUEST. Damit erhalten alle Benutzer, die die Anwendung mit AutoLogon=True (siehe <Anwendungsname>.ini) starten, ohne Authentifizierung die Rolle GUEST zugeordnet.

3.4.1.2. Browser-Verzeichnis

Der bereits mehrfach erwähnte cyAgent-Browser (Abb. 5) stellt dem Anwender nach dem Start des cyAgent und der Eingabe einer gültigen cyProvider-URL zunächst eine Übersicht aller verfügbaren Anwendungen zur Verfügung. Es werden nur die Anwendungen angezeigt, für die im Verzeichnis Browser\Def-Files eine Datei <Anwendungsname>.ini angelegt wurde, in der der Name der Anwendung und das Logon-Verfahren definiert sind. In den weiteren Unterverzeichnissen Images und RTF sind Kurztexte zur Beschreibung der Anwendungen und Bitmaps als Anwendungslogo hinterlegt. Durch Anklicken des Logos beginnt dann die Authentifizierung und der anschließende Aufbau der Anwendung.

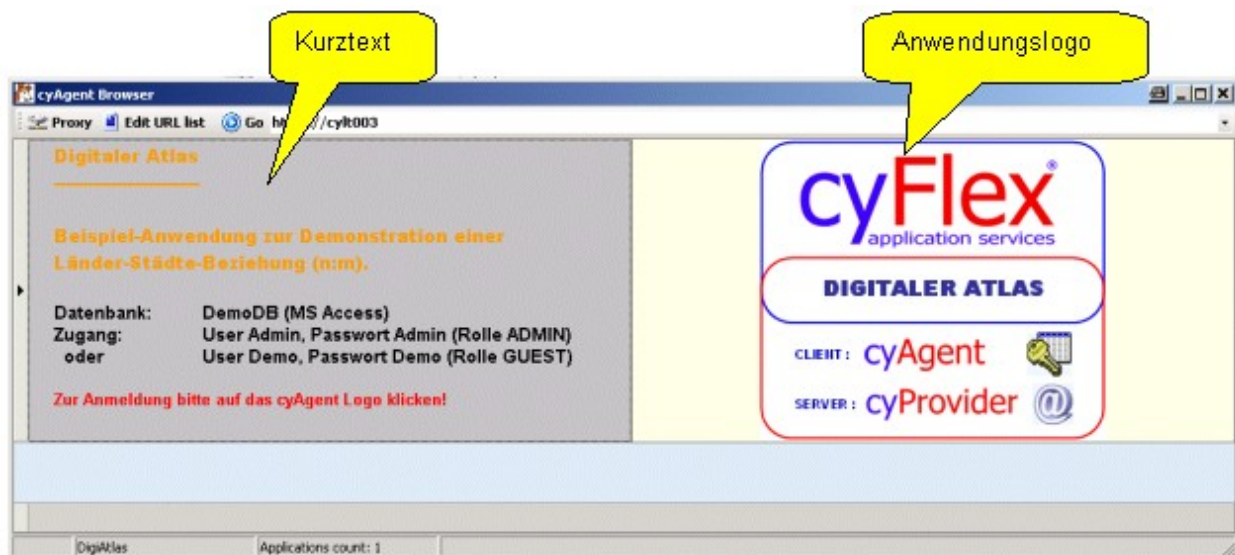


Abb. 5

3.4.1.3. Globals-Verzeichnis

Das Verzeichnis Globals ist für allgemein gültige Einträge reserviert.

Im Unterverzeichnis Language befinden sich die anwendungsübergreifenden Sprachdefinitionen, die je Sprache in Dateien mit dem Namen <countrycode>_Globals.cdd abgelegt sind. Neben System-/Fehlermeldungen, Beschriftungen der Buttons, Hinweistexten zu den Icons, Beschriftungen des Stylesheet-Editors oder der Darstellung von Kalenderobjekten (Tage, Monate) sind alle globalen Sprachdefinitionen in diesen Dateien enthalten.

3.4.2. Images-Verzeichnis

Die oben beschriebenen Dateien des Verzeichnisses Sources können mit dem Programm cyImager zur Verbesserung der Download-Performance und der initialen Ladezeit in den Hauptspeicher vorkompiliert und in dem Verzeichnis Images abgelegt werden. Damit die Dateien aus dem Image-Verzeichnis anstelle derer aus dem Source-Verzeichnis zum cyAgent geschickt werden, muss der Parameter UseImage=y in der *cyProvider.ini* gesetzt sein. Für jede vorkompilierte Anwendung wird ein eigenes Unterverzeichnis in Images angelegt. Sollen Images anstelle der Sources eingesetzt werden, müssen alle Anwendungen und das oben beschriebene Browser-Verzeichnis vorkompiliert werden.

3.5. Aufruf von Anwendungen mit dem cyAgent

Es gibt mehrere Möglichkeiten eine Anwendung mit dem cyAgent zu starten:

- Sind im cyAgent-Verzeichnis keine ini-Dateien vorhanden, startet der cyAgent zunächst den cyAgent-Browser, um sich im ersten Schritt die für die Anwendung erforderliche ini-Datei vom cyProvider zu beschaffen.
- Findet der cyAgent in seinem Verzeichnis eine *cyGui.ini* Datei, versucht er die darin angegebene Anwendung direkt, ohne den Umweg über den cyAgent-Browser, zu starten.
- Existiert auf Client-Seite im cyAgent-Verzeichnis eine ini-Datei mit dem Namen der Anwendung, lässt sich der direkte Aufruf auch über ein Desktop-Icon realisieren, indem der Name der Anwendung als Parameter übergeben wird (z.B. C:\cyAgent\cyAgent.exe DigiAtlas).

3.6. Übersicht der zu administrierenden Dateien

Abschließend sind die für den Administrator relevanten Dateien am Beispiel der Beispielanwendung Digitaler Atlas noch einmal zusammengestellt:

- *\cyProvider\conf.ini*
- *\cyProvider\cyProvider.ini*
- *\cyProvider\Sources\Browser\DEF-FILES\DigiAtlas.ini*
- *\cyProvider\Sources\DigiAtlas\DEF-FILES\cyGuiApl.ini*
- *\cyProvider\Logs\cyProviderTicket.log*
- *\cyProvider\Logs\cyProviderTrace.log*

