

IBM ACS – Plugins Teil 2

Dieser TechKnowLetter schließt an “[IBM i ACS – Plug-ins und SQL](#)” und “[IBM i ACS – Datentransfer und Plug-ins](#)” an, in dem die Datenbank-PlugIns beschrieben wurden. **ABER** da gibt es mehr!

1. Anmeldung an ein System

Empfehlenswert ist vor der Verwendung von ACS Plug-Ins sich generell erstmal an dem die Aktionen ausführenden System anzumelden, wobei dieses, wie in ACS üblich, in einem Cache gespeichert werden:

```
java -jar acsbundle.jar /PLUGIN=logon /SYSTEM=<IBM i System> /USERID=<USERNAME> /AUTH
```

```
***** Signon to IBM i (<IBM i System>)
```

```
User (<USERNAME>):
```

```
Password:
```

```
MSGGEN002 - The function completed successfully.
```

2. Löschung der gespeicherten Anmeldeinformationen

Praxisrelevant ist auch diese Anmeldeinformationen zu löschen, um sich mit einem anderen User zu verbinden:

```
java -jar acsbundle.jar /PLUGIN=MAINT /CLEARPWCACHES
```

Möchte man solche Befehle skripten, also durch eine Art Programm ausführbar machen, könnte dies in Windows z.B. über eine Batchdatei eingerichtet werden. Dazu erstelle man eine Datei mit der Extension *.bat* mit folgendem Inhalt

```
@echo off
OLDPATH=%CD%
cd <ACS directory>
start /b javaw -jar acsbundle.jar /PLUGIN=MAINT /CLEARPWCACHES
cd %OLDPATH%
```

und speichere diese mit der Extension *.bat* ab. (Hinweis: Beim Speichern unter einer anderen Dateierstension als *txt* erstmal von der Auswahl **.txt* auf alle Dateien, also **.**, wechseln!)

Erläuterung des Ablaufs: Nach Abschaltung der Ausgaben durch *@echo off* wird das aktuelle Verzeichnis in *OLDPATH* gespeichert und anschließend in das ACS Verzeichnis gesprungen, wo sich die Datei *acsbundle.jar* befindet, aus der das Plug-In gestartet wird. Am Ende des Batchprogramms wird in das ursprüngliche Verzeichnis zurückgewechselt. Der Befehl *javaw* startet wie *java* ein Java-Programm, öffnet aber dafür kein eigenes Fenster.

Mit Hilfe der Option */D* des *start* Befehls kann alles auch in einer Zeile ausgeführt werden:

```
start /B /D <ACS directory> javaw -jar acsbundle.jar /PLUGIN=MAINT /CLEARPWCACHES
```

3. Verbindungsprüfung

Im Windows Paket von ACS findet man den altbekannten Befehle cwbping, der eine Überprüfung der Erreichbarkeit von Servern der jeweiligen IBM i Instanz durchführt. Im Linux oder in PASE gab es so einen Befehl nie, aber man kann sich mit einem ACS Plug-In ähnliche Funktionalität schaffen:

```
java -jar acsbundle.jar /PLUGIN=ping /SYSTEM=<IBM i System>
```

```
Verifying connection to port mapper service...Success! using port number 449
Verifying connection to central server service...Success! using port number 8470
Verifying connection to command service...Success! using port number 8475
Verifying connection to database service...Success! using port number 8471
Verifying connection to data queues service...Success! using port number 8472
Verifying connection to file service...Success! using port number 8473
Verifying connection to print service...Success! using port number 8474
Verifying connection to record-level access service...Success! using port number 446
Verifying connection to signon service...Success! using port number 8476
Verifying connection to Navigator for i service...Success! using port number 2002
Verifying connection to Secure Shell (SSH) service...Success! using port number 22
Verifying connection to Telnet service...Success! using port number 23
Done
```

Es besteht auch die Möglichkeit damit SSL Verbindungen, die aus Security Gründen vorzuziehen, zu prüfen. Dazu genügt es noch die Option `/SSL=1` hinzuzufügen. Der Default ist `/SSL=0`. Im nachfolgenden Beispiel wurde SSL nicht konfiguriert, wie man aus der Ausgabe unschwer ersehen kann.

```
java -jar acsbundle.jar /PLUGIN=ping /SYSTEM=<IBM i System> /SSL=1
```

Verifying connection to port mapper service...Success! using port number 449

Verifying connection to central server service...Failed **** MSGSOCK007 - Could not connect. (Connection refused: connect) using port number 9470

Verifying connection to command service...Failed **** MSGSOCK007 - Could not connect. (Connection refused: connect) using port number 9475

Verifying connection to database service...Failed **** MSGSOCK007 - Could not connect. (Connection refused: connect) using port number 9471

Verifying connection to data queues service...Failed **** MSGSOCK007 - Could not connect. (Connection refused: connect) using port number 9472

Verifying connection to file service...Failed **** MSGSOCK007 - Could not connect. (Connection refused: connect) using port number 9473

Verifying connection to print service...Failed **** MSGSOCK007 - Could not connect. (Connection refused: connect) using port number 9474

Verifying connection to record-level access service...Failed **** MSGGEN004 - An unexpected end of the file or stream has been encountered. (SSL peer shut down incorrectly) using port number 448

Verifying connection to signon service...Failed **** MSGSOCK007 - Could not connect. (Connection refused: connect) using port number 9476

Verifying connection to Navigator for i service...Failed **** MSGSSL001 - An error was encountered during a secure socket operation. (Unsupported or unrecognized SSL message) using port number 2002

Verifying connection to Secure Shell (SSH) service...Success! using port number 22

Verifying connection to Telnet service...Failed **** MSGSOCK007 - Could not connect. (Connection refused: connect) using port number 992

Done

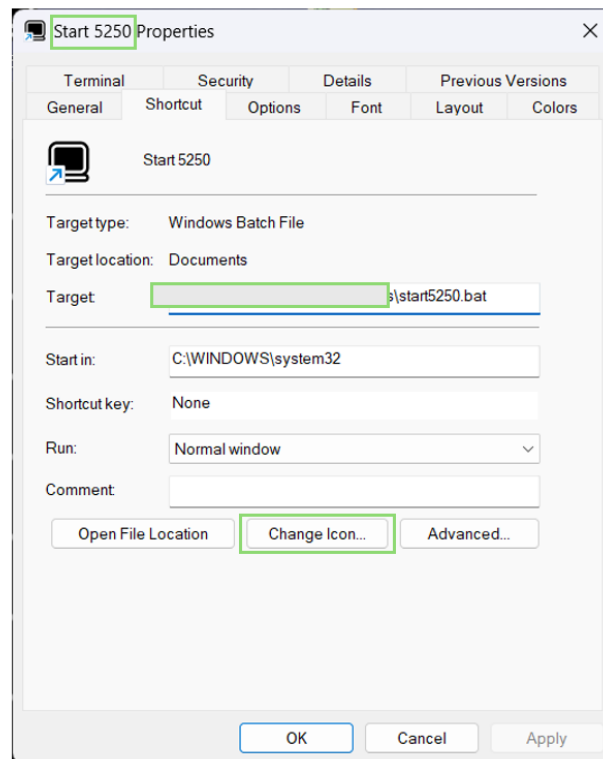
4. Starten einer 5250 Session

Ich empfehle dazu wieder den *start* Befehl von Windows und *javaw* zu verwenden, um das Öffnen eines weiteren Terminalfensters zu vermeiden:

start /b javaw -jar acsbundle.jar /PLUGIN=5250 /SYSTEM=<IBM i System>

Name des Links

Name des Batch
Programms



Ähnlich wie beim Leeren des Cache kann man diesen Aufruf in einer Batchdatei hinterlegen, einen Link dazu auf den Desktop erstellen, diesen umbenennen und letztendlich über die Eigenschaften des Links ihm auch noch ein eigenes Icon zuordnen.

Damit hat man alle Verbindungen zu Systemen, die man nutzen möchte auf seinem Desktop und kann diese ohne Aufruf der ACS GUI einzeln starten.

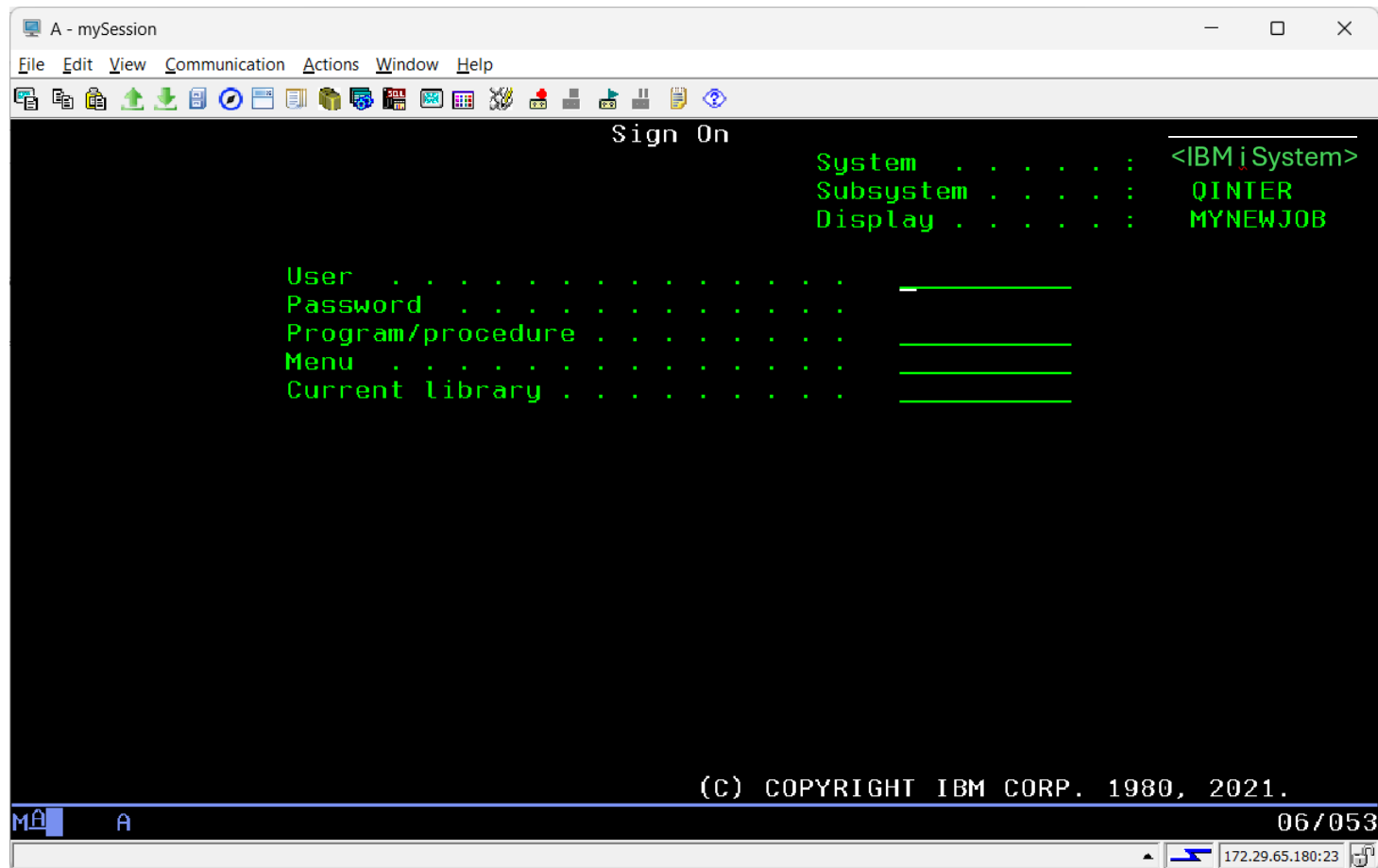
Dieses, wie auch die anderen Plug-Ins haben Zusatzoptionen, die eine individuelle Konfiguration eines 5250 Terminalfensters ermöglichen.

Verfügbar für das 5250 Plug-In sind:

/id=<A-Z>	- Kurze Sitzungs-ID
/name=<name>	- Sitzungsname
/wsid=<kennung>	- Workstation-ID
/wide=<1 0 wahr falsch>	- Breite Anzeige verwenden (27x132)
/fullscreen<1 0 wahr falsch>	- Gesamte Anzeige verwenden
/nosave=<1 0 wahr falsch>	- Einstellungen beim Beenden nicht speichern
/prompt=<1 0 wahr falsch>	- Aufruf des Konfigurationsdialogfensters erzwingen
/port=<port>	- Portnummer
/ssl=<1 0 wahr falsch>	- Verbindung über Secure Sockets herstellen
/sso=<1 0 wahr falsch>	- Anmeldeanzeige umgehen
/kerberos	- Kerberos verwenden
/width=<breite>	- Ausgangsbreite des Emulatorfensters
/height=<höhe>	- Ausgangshöhe des Emulatorfensters
/xpos=<x-pos>	- Ausgangsposition der X-Koordinate für die linke obere Ecke des Emulatorfensters
/ypos=<ypos>	- Ausgangsposition der Y-Koordinate für die linke obere Ecke des Emulatorfensters
/watermark	- Systemname als Wasserzeichen auf Bildschirm anzeigen
/watermark	- Angegebenen Text als Wasserzeichen auf Bildschirm anzeigen

Zum Beispiel wird ein 5250 Terminalemulation mit Namen *mySession* und der Workstation-ID *myNewJob* startet mit:

start /b javaw -jar acsbundle.jar /PLUGIN=5250 /SYSTEM=<IBM i System> /name=mySession /wsid=myNewJob /wide=1



```
java -jar acsbundle.jar /PLUGIN=cldownload /SYSTEM=<IBM i System> /hostfile=QIWS/QCUSTCDT /DISPLAY
```

CUSNUM	LSTNAM	INIT	STREET	CITY	STATE	ZIPCOD	CDTLMT	CHGCOD	BALDUE	CDTDUE
938472	Henning	G K	4859 Elm Ave	Dallas	TX	75217	5000	3	37.00	0.00
839283	Jones	B D	21B NW 135 St	Clay	NY	13041	400	1	100.00	0.00
392859	Vine	S S	PO Box 79	Broton	VT	5046	700	1	439.00	0.00
938485	Johnson	J A	3 Alpine Way	Helen	GA	30545	9999	2	3987.50	33.50
397267	Tyron	W E	13 Myrtle Dr	Hector	NY	14841	1000	1	0.00	0.00
389572	Stevens	K L	208 Snow Pass	Denver	CO	80226	400	1	58.75	1.50
846283	Alison	J S	787 Lake Dr	Isle	MN	56342	5000	3	10.00	0.00
475938	Doe	J W	59 Archer Rd	Sutter	CA	95685	700	2	250.00	100.00
693829	Thomas	A N	3 Dove Circle	Casper	WY	82609	9999	2	0.00	0.00
593029	Williams	E D	485 SE 2 Ave	Dallas	TX	75218	200	1	25.00	0.00
192837	Lee	F L	5963 Oak St	Hector	NY	14841	700	2	489.50	0.50
583990	Abraham	M T	392 Mill St	Isle	MN	56342	9999	3	500.00	0.00

Export to file:

```
java -jar acsbundle.jar /PLUGIN=cldownload /SYSTEM=<IBM i System> /hostfile=QIWS/QCUSTCDT
/clientfile=C:\Users\119633724\acs_download.xlsx
```

Ad-hoc select:

```
java -jar acsbundle.jar /PLUGIN=cldownload /SYSTEM=<IBM i System> /sql="select LSTNAM,STREET from QIWS/QCUSTCDT" /DISPLAY
```

LSTNAM	STREET
Henning	4859 Elm Ave
Jones	21B NW 135 St
Vine	PO Box 79
Johnson	3 Alpine Way
Tyron	13 Myrtle Dr
Stevens	208 Snow Pass
Alison	787 Lake Dr
Doe	59 Archer Rd
Thomas	3 Dove Circle
Williams	485 SE 2 Ave
Lee	5963 Oak St
Abraham	392 Mill St

```
javaw -jar acsbundle.jar /PLUGIN=SPLF /SYSTEM=<IBM i System>h
```

```
java -jar acsbundle.jar /PLUGIN=backup
```

Creating zip file 'C:\Users\119633724\Documents\IBM\iAccessClient_Service\acs_bak.zip'

Adding: cacerts

Adding: Plugins/console_com.ibm.iaccess.oc.plugins.x0041csx0053systemx0043onsolex0050login_version_2.0/profiles.zip

Adding: Plugins/rss_com.ibm.iaccess.plugins.db.runsql.x0052unx0053qlx0053criptsx0050lugin_version_2.0/rss.zip

Adding: Plugins/sm_com.ibm.iaccess.hod.plugins.x0041csx0047lobalx0048odx0050lugin_version_2.0/config.zip

Adding: Plugins/sm_com.ibm.iaccess.hod.plugins.x0041csx0047lobalx0048odx0050lugin_version_2.0/dtRequests.zip

Adding: Plugins/sm_com.ibm.iaccess.hod.plugins.x0041csx0047lobalx0048odx0050lugin_version_2.0/profiles.zip

Adding: settings.xml