SAS 9 i zewnętrzne źródła danych

Zespół Wsparcia Technicznego, SAS Institute Polska czerwiec 2019

Szanowni Państwo!

Celem niniejszego dokumentu jest przedstawienie podstawowych zagadnień związanych z konfiguracją SAS do pracy z zewnętrznymi źródłami danych, takimi jak Excel czy systemy bazodanowe (Oracle, SQL Server, Teradata itp.). Poruszone zostały m.in. następujące tematy:

- Wymagania licencyjne SAS dla dostępu do zewnętrznych źródeł danych,
- Konfiguracja SAS w celu dostępu do danych,
- Obsługa polskich znaków.

Niniejszy dokument oraz inne informacje związane z tematem będą publikowane na naszych stronach WWW pod adresem http://www.sas.com/pl_pl/support/dokumenty.

W razie dodatkowych pytań prosimy o kontakt z naszym Zespołem, telefonicznie na numer **(22) 560 4666** lub elektronicznie pod adresem support@spl.sas.com.

Prosimy o uwagi i spostrzeżenia, które pomogą nam w przygotowaniu następnych wersji tego dokumentu.

Zawsze służąc Państwu pomocą,
Zespół Pomocy Technicznej
Centrum Technologii SAS Polska

Spis treści

1 Informacje ogólne 4
Weryfikacja licencji 4
Weryfikacja instalacji modułu SAS/ACCESS 4
Instalacja klienta bazy danych 5
Polskie znaki oraz Unicode 5
Ustawianie zmiennych środowiskowych 7
WINDOWS 7
LINIX / UNIX / AIX 9
Weryfikacja wartości zmiennych środowiskowych 9
2 Pliki tekstowe 11
Wymagania 11
Konfiguracja 11
Użycie 11
Polskie znaki 12
3 XML 13
Wymagania 13
Konfiguracja 13
Użycie 13
Polskie znaki 14
4 PC Files (MS Excel, MS Access) 15
Wymagania 15
Konfiguracja 15
Użycie 15
Polskie znaki 16
5 Oracle 17
Wymagania 17
Konfiguracja 17
Polskie znaki 18
6 ODBC 19
Wymagania 19
Konfiguracja 19
Windows 19
UNIX 20
Polskie znaki 23
7 SQL Server 24
Wymagania 24
Konfiguracja 24
Windows 24
UNIX 24
Polskie znaki 25
8 MySQL 26
Wymagania 26
Konfiguracja 26
Polskie znaki 26
9 PostgreSQL 28
Wymagania 28

Konfiguracja 28
Polskie znaki 28
Windows 29
UNIX 29
10 Teradata 30
Wymagania 30
Konfiguracja 30
Klient 30
Teradata Parallel Transporter (TPT) 30
FastExport 31
MultiLoad 31
Polskie znaki 32
11 Hadoop 33
Wymagania 33
Konfiguracja 33
Polskie znaki 34
12 Dodatkowe informacje 35

1 Informacje ogólne

SAS domyślnie pozwala na odczyt danych z plików tekstowych, plików JMP oraz własnych formatów danych, takich jak sas7bdat.

Dostęp do danych z innych źródeł wymaga zakupu licencji dla odpowiedniego modułu SAS/ACCESS (np. SAS/ACCESS Interface to PC Files lub SAS/ACCESS Interface to Oracle). W przypadku baz danych, zazwyczaj konieczna jest również instalacja i konfiguracja klienta danej bazy.

Weryfikacja licencji

Aby sprawdzić, jakie źródła danych są obsługiwane przez daną instalację SAS-a, należy najpierw zweryfikować zawartość aktualnej licencji. W tym celu należy wykonać poniższy kod:

```
proc setinit;
run;
```

Po wykonaniu kodu, w logu powinniśmy otrzymać zawartość aktualnej licencji SAS. Poniżej przedstawiany został przykładowy rezultat:

```
Pierwotne dane instalacji
Bieżąca wersja: 9.04.01M6P110718
Nazwa instalacji:   'XXXXXXXXXXXXXXXXXXX'.
Numer instalacji:   XXXXXXXX.
CPU A: nazwa modelu='' numer modelu='' seria=''.
Data wygaśnięcia:   30MAY2020.
Okres karencji:      45 dni (zakończenie 14JUL2020).
Okres ostrzegawczy:  47 dni (zakończenie 30AUG2020).
Data zainicjowania:  23MAY2019.
System operacyjny:   WX64_E  .
Daty wygaśnięcia produktów:
---Oprogramowanie Base SAS
     30MAY2020 (CPU A)
---SAS/STAT
     30MAY2020 (CPU A)
---SAS/GRAPH
     30MAY2020 (CPU A)
---SAS/ACCESS Interface to PC Files
     30MAY2020 (CPU A)
---SAS/ACCESS Interface to ODBC
     30MAY2020 (CPU A)
```

W przedstawionym przykładzie SAS posiada licencję na moduły SAS/ACCESS Interface to PC Files oraz SAS/ACCESS Interface to ODBC.

Jeżeli w wyniku nie ma odpowiedniego modułu SAS/ACCESS, należy zaaplikować nową licencję, zawierającą ten moduł. Proces aktualizacji licencji został opisany w dokumencie Aktualizacja licencji oprogramowania SAS.

Weryfikacja instalacji modułu SAS/ACCESS

Oprócz ważnej licencji, do poprawnego działania modułu SAS/ACCESS konieczna jest jego poprawna instalacja. W zdecydowanej większości przypadków podczas instalacji SAS-a instalowane są wszystkie licencjonowane moduły. Możliwa jest jednak sytuacja, w której z jakichś powodów lista zainstalowanych modułów nie pokrywa się z listą w licencji.

W takiej sytuacji listę zainstalowanych modułów można sprawdzić wykonując poniższy kod:

```
proc product_status;
run;
```

W rezultacie w logu powinniśmy otrzymać listę zainstalowanych modułów, wraz z wersjami.

```
For Oprogramowanie Base SAS ...
   Custom version information: 9.4_M6
   Image version information: 9.04.01M6P110718
For SAS/STAT ...
   Custom version information: 15.1
For SAS/GRAPH ...
   Custom version information: 9.4_M6
For SAS/ACCESS Interface to PC Files ...
   Custom version information: 9.4_M6
For SAS/ACCESS Interface to ODBC ...
   Custom version information: 9.4_M6
```

W przypadku rozbieżności pomiędzy licencją a listą zainstalowanych komponentów, zalecamy usunięcie i ponowną instalację SAS Foundation. Procesu instalacji SAS można znaleźć w dokumencie Instalacja SAS 9.4 Foundation i SAS Enterprise Guide.

Instalacja klienta bazy danych

Aby SAS był w stanie podłączyć się do bazy danych, w większości przypadków na serwerze musi być zainstalowany i skonfigurowany klient tej bazy. W zależności od wersji SAS-a, wspierane są różne wersje klientów.

Szczegółowe instrukcje na temat wspieranych wersji klientów znajdują się w dokumentach System Requirements. Bezpośrednie linki do najpopularniejszych wersji znajdują się w poniższej tabelce:

	WINDOWS	LINUX	AIX
SAS 9.2	32 BIT, 64 BIT	32 BIT, 64 BIT	64 BIT
SAS 9.3	32 BIT, 64 BIT	32 BIT, 64 BIT	64 BIT
SAS 9.4	32 BIT, 64 BIT	64 BIT	64 BIT

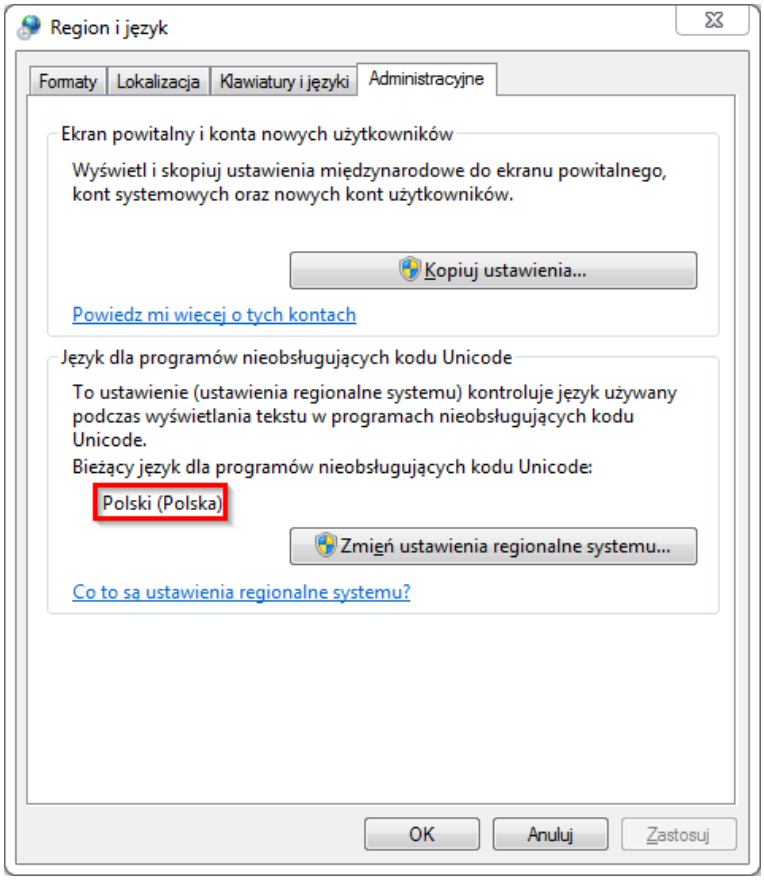
Szczegółowe informacje na temat wspieranych wersji klientów dla poszczególnych baz danych znajdą Państwo również w dalszych rozdziałach tego dokumentu.

Polskie znaki oraz Unicode

W przypadku systemów Windows i problemów z widocznością polskich znaków pochodzących z zewnętrznych źródeł danych (plik na dysku, baza danych) proponujemy zweryfikować ustawienia zgodnie z notą http://support.sas.com/kb/18/849.html. Nota ta, mimo iż napisana dla SAS-a w wersji 9.1, jest w dalszym ciągu aktualna. Aktualnym odpowiednikiem dokumentu wskazywanego przez notę jest SAS 9.4 National Language Support (NLS): Reference Guide.

Szczególnie częstą przyczyną problemów z obsługą polskich znaków jest niepoprawna wartość parametru *„Język dla programów nieobsługujących kodu Unicode” (Language for Non-unicode programs)* w systemie Windows. Ustawienie to można zmienić w następującej lokalizacji: *Panel Sterowania → Zegar,*

język i region → Region i język → Administracyjne. Parametr powinien mieć wartość Polski (Polska). Poprawne ustawienie zostało przedstawione na poniższym zrzucie ekranu.



Oprócz ustawień na poziomie systemowym, wiele klientów baz danych wymaga ustawienia dodatkowych opcji SAS-owych oraz zmiennych środowiskowych. Najczęstsze wymagania zostały przedstawione w poniższych rozdziałach. Szerszy opis zagadnień związanych z przesyłaniem danych pomiędzy SAS-em a bazami danych można znaleźć w dokumencie Multilingual Computing with SAS 9.4.

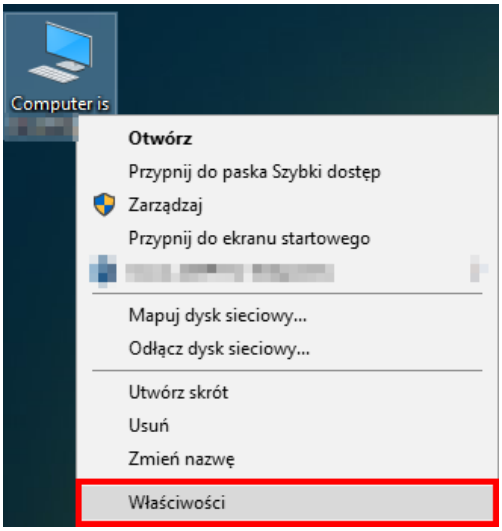
Ustawianie zmiennych środowiskowych

Proces konfiguracji modułów SAS/ACCESS często wymaga ustawienia odpowiednich zmiennych środowiskowych w systemie operacyjnym. W tym rozdziale przedstawiamy przykładowy proces ustawiania takich zmiennych.

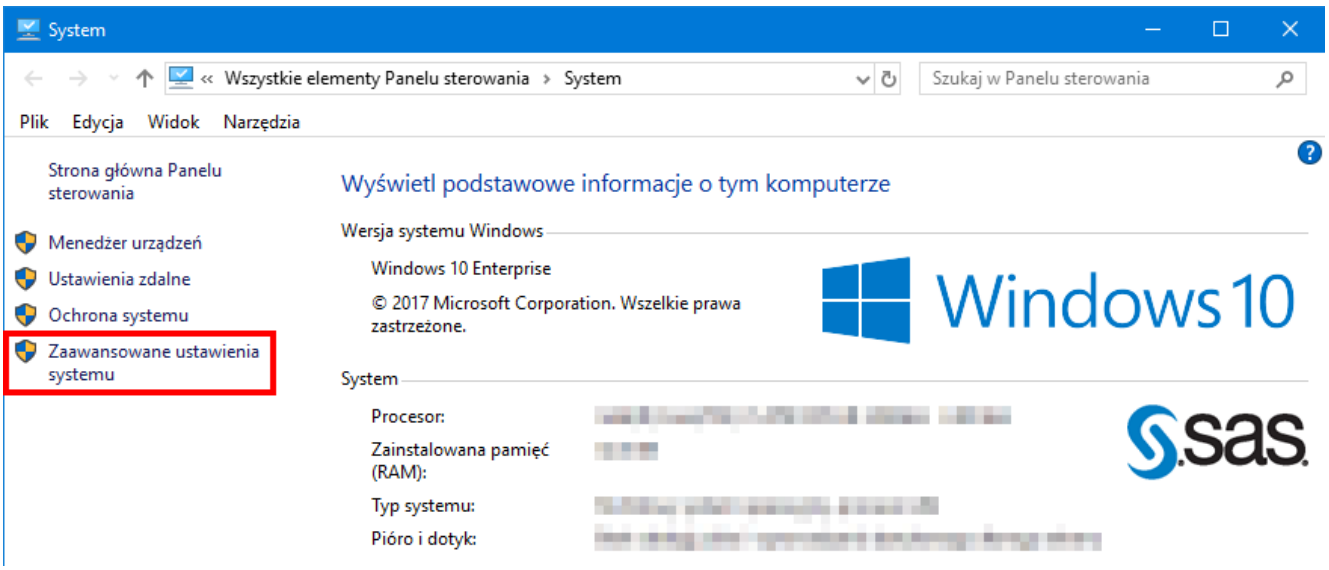
WINDOWS

W systemach Windows oraz Windows Server zmienne środowiskowe ustawia się w następujący sposób:

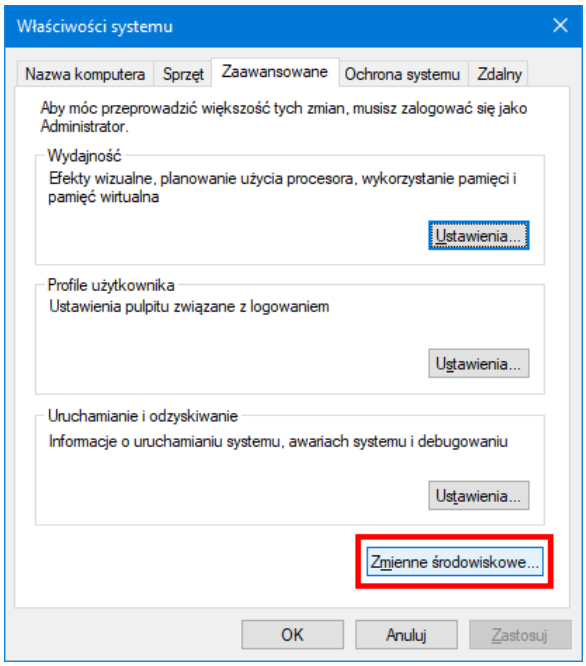
1. Otwieramy właściwości systemu (**Panel Sterowania -> System** albo klikamy prawym przyciskiem myszy na ikonę komputera na pulpicie i wybieramy Właściwości).



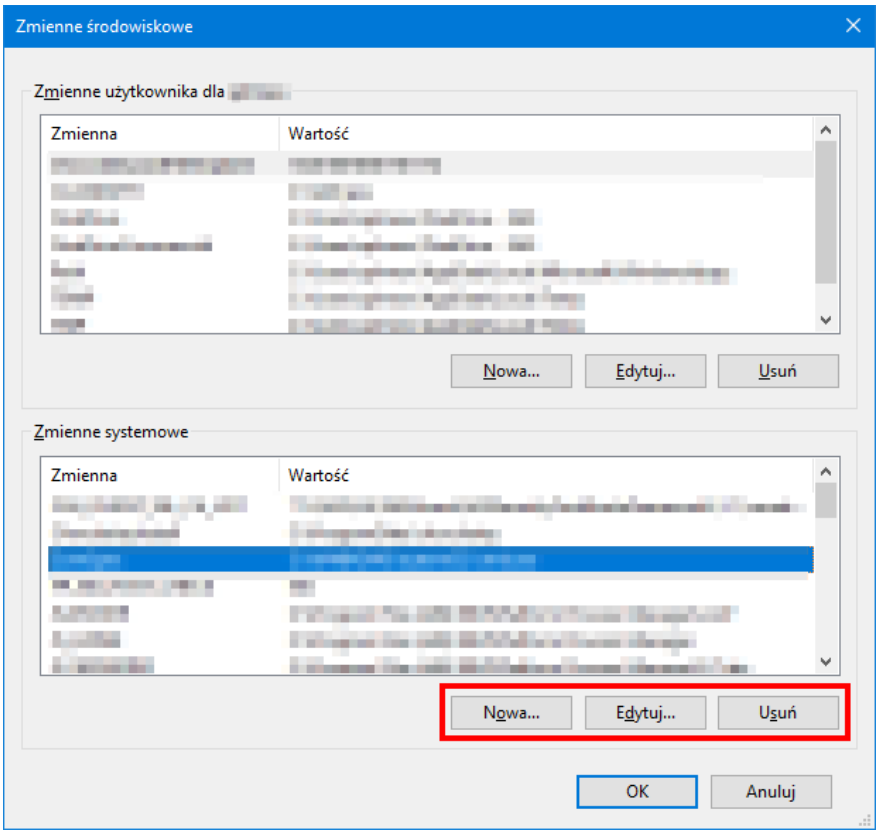
2. W oknie właściwości systemu wybieramy **Zaawansowane ustawienia systemu**.



3. W kolejnym oknie klikamy przycisk **Zmienne środowiskowe...**



4. W zależności od wykonywanej operacji w sekcji **Zmienne systemowe** klikamy przycisk **Nowa…**, **Edytuj…** bądź **Usuń** i ustawiamy odpowiednią zmienną.



Wprowadzone zmiany są zaczytywane przy następnym uruchomieniu aplikacji (np. edytor SAS BASE). W przypadku usług SAS (np. Metadata Server czy Object Spawner) wymagany jest ich restart.

LINIX / UNIX / AIX

Na platformach innych niż Windows, zalecaną metodą ustawiania zmiennych środowiskowych dla SAS-a jest edycja pliku ***<SAS_HOME>/SASFoundation/9.X/bin/sasenv_local***.

Aby sprawdzić istniejące zmienne środowiskowe należy uruchomić terminal i wykonać polecenie **set**.

Aby zdefiniować nową zmienną środowiskową dla SAS w pliku **sasenv_local** należy dodać nową linię według schematu

```
export <NAZWA_ZMIENNEJ>=<WARTOŚĆ_ZMIENNEJ>
```

Aby dopisać nową wartość do zmiennej (dotyczy to głównie zmiennych PATH oraz LD_LIBRARY_PATH / LIBPATH / SHLIB_PATH) należy dopisać linię według schematu

```
export <NAZWA_ZMIENNEJ>=<NOWA_WARTOŚĆ>:$<NAZWA_ZMIENNEJ>
```

Powyższy przykład dopisuje nową wartość na początku zmiennej.

Weryfikacja wartości zmiennych środowiskowych

Aby zweryfikować czy zmienne środowiskowe są ustawione zgodnie z oczekiwaniami, można użyć poniższego kodu:

```
filename env pipe "set";

data _null_;
    infile env;
    input;
    put _infile_;
run;
```

W rezultacie w logu powinniśmy otrzymać listę wszystkich zmiennych środowiskowych oraz ich aktualnych wartości.

Do poprawnego działania powyższego kodu wymagane jest, aby SAS mógł wykonywać polecenia w systemie operacyjnym (musi być ustawiona opcja XCMD). W przeciwnym wypadku próba uruchomienia programu zwróci poniższy błąd:

```
1    filename env pipe "set";
ERROR: Insufficient authorization to access PIPE.
ERROR: Error in the FILENAME statement.
```

W takiej sytuacji możemy jedynie sprawdzać wartości pojedynczych zmiennych środowiskowych. W tym celu można użyć poniższy kod:

```
data _null_;
   length val $1024;
   val = sysget("<NAZWA_ZMIENNEJ>");
   put val=;
run;
```

2 Pliki tekstowe

Wymagania

BRAK - odczyt i zapis plików tekstowych jest integralną funkcjonalnością SAS-a i nie wymaga dodatkowych modułów czy oprogramowania firm trzecich.

Konfiguracja

Dostęp do plików tekstowych nie wymaga żadnych kroków konfiguracyjnych.

Użycie

Bardzo często dane w plikach tekstowych przechowywane są w następujący sposób:

- Pierwsza linia w pliku zawiera nazwy kolumn.
- Każda kolejna linia zawiera wartości komórek z pojedynczego wiersza tabeli
- Poszczególne wartości w linii rozdzielone są określonym separatorem (np. spacją, przecinkiem czy średnikiem)

Pliki spełniające powyższe założenia nazywamy plikami z wartościami rozdzielanymi (ang. delimited text file). Pliki w tym formacie można importować i eksportować używając procedur ***PROC IMPORT*** oraz ***PROC EXPORT***. Oto przykładowy kod:

```
/*Eksport danych do pliku tekstowego*/
proc export outfile='C:\class.txt'      /*plik wynikowy*/
       data=sashelp.class  /*eksportowane dane*/
       dbms=dlm                  /*typ wyniku - plik z wart. rozdzielanymi separatorem*/
       replace;                  /*nadpisujemy plik jeżeli istnieje*/
       delimiter=';';            /*znak separatora,  w tym przypadku średnik*/
run;

/*Import danych z powrotem do SAS-a*/
proc import datafile='C:\class.txt'     /*plik źródłowy*/
       out=work.class            /*tabela docelowa*/
       dbms=dlm                  /*typ pliku*/
       replace;                  /*nadpisujemy tabelę jeżeli istnieje*/
       delimiter=';';            /*znak separatora*/
run;
```

Utworzony plik wygląda następująco:

```
Name;Sex;Age;Height;Weight
Alfred;M;14;69;112.5
Alice;F;13;56.5;84
Barbara;F;13;65.3;98
...
```

Drugim bardzo popularnym formatem plików tekstowych są pliki z kolumnami o stałej szerokości. W takich plikach wartości poszczególnych kolumn zawsze znajdują się na określonych pozycjach (np. pierwsza wartość to znaki 1-10, druga to znaki 12-15 itd.).

W przypadku takich plików oraz plików o niestandardowym formacie odczyt oraz zapis danych odbywa się przy użyciu DATA STEP-u oraz instrukcji ***INPUT*** oraz ***PUT***. Instrukcje te pozwalają zaimportować oraz wygenerować pliki tekstowe o dowolnej strukturze. Dokładny opis ich działania oraz dostępnych opcji znajduje się w dokumentacji SAS (INPUT, PUT). Dobre źródło wiedzy stanowi również dokument The SAS INFILE and FILE Statements.

Dla celów pokazania użycia instrukcji ***PUT*** oraz ***INPUT*** spróbujemy utworzyć plik tekstowy z kolumnami o stałej szerokości znaków a następnie odczytać jego zawartość.

```
filename txtfile 'C:\SAS\class_2.txt';

/*eksport do pliku*/
data _null_;
       set sashelp.class;  /*dane do ekportu*/
       file txtfile;       /*plik wynikowy*/
       /*format wyekportowanych danych (stałopozycyjny)*/
       put name 1-8 sex 10 age 12-13 height 15-17 weight 19-21;
run;

/*import z pliku*/
data work.class2;
       infile txtfile;           /*plik źródłowy*/
       /*format importowanych danych*/
       input name $ 1-8 sex $ 10 age 12-13 height 15-17 weight 19-21;
run;
```

Utworzony plik tekstowy ma w tym przypadku następującą postać:

```
Alfred   M 14   69 113
Alice    F 13   57  84
Barbara  F 13   65  98
Carol    F 14   63 103
...
```

Polskie znaki

Podczas operacji odczytu i zapisu danych do pliku SAS domyślnie stosuje kodowanie znaków swojej sesji. Może to powodować problemy z odczytem polskich znaków, jeżeli plik stosuje inne kodowanie niż aktualne kodowanie sesji SAS-owej, bądź gdy SAS działa w kodowaniu nie obsługującym polskich znaków.

Aby uniknąć tego problemu plik tekstowy należy wskazać przy pomocy instrukcji ***FILENAME***. Instrukcja ta posiada opcję **ENCODING=**, pozwalającą wskazać jego kodowanie. Kodowanie obsługujące polskie znaki to WLATIN2 (Windows) oraz LATIN2 (Linux/Unix).

Przykładowe wykonanie:

```
filename txtpol 'C:\polskie.txt' encoding=wlatin2;

proc import datafile=txtpol
       out=work.polskie
       dbms=csv
       replace;
run;
```

3 XML

Wymagania

BRAK – obsługa plików XML (LIBNAME XML) jest wbudowaną funkcjonalnością SAS i nie wymaga żadnych dodatkowych modułów.

Konfiguracja

BRAK.

Użycie

SAS dostarcza 2 metody dostępu do plików XML – LIBNAME XML (dostępna od wersji 9.1.3) oraz LIBNAME XMLV2 (zwana również XML92, dostępna od wersji 9.2). LIBNAME XMLV2 jest rozwinięciem LIBNAME XML i jest z nią wstecznie kompatybilna

Funkcjonalności dodane w silniku XMLV2 to m.in.:

- Obsługa wielu tabel w 1 pliku – LIBNAME XML pozwala przechowywać w pojedynczym pliku dane z tylko 1 tabeli
- Obsługa map XML – pozwalają na import i eksport plików XML o formatach innych niż domyślny

Domyślny format dokumentu XML odczytywanego i tworzonego przez SAS jest następujący:

- Pierwsza linia zawiera nagłówek z informacją o wersji oraz kodowaniu znaków
- Wszystkie dane są zapisane w sekcji <TABLE>
- Poszczególne wiersze tabeli są zapisane w sekcjach o nazwie odpowiadającej nazwie tabeli
- Wartości poszczególnych kolumn w ramach wiersza są zapisane w sekcjach o nazwach odpowiadających poszczególnym kolumnom

Poniżej przedstawiony został przykładowy kod eksportujący tabelę do pliku XML oraz zawartość wygenerowanego pliku:

```
libname xmllib xml path="C:\SAS\class.xml";

data xmllib.class;
    set sashelp.class;
run;
```

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<TABLE>
   <CLASS>
      <Name>Alfred</Name>
      <Sex>M</Sex>
      <Age>14</Age>
      <Height>69</Height>
      <Weight>112.5</Weight>
   </CLASS>
...
</TABLE>
```

Aby zaczytać bądź utworzyć plik XML w innym formacie, należy stworzyć mapę XML. W przypadku importu pliku możemy posłużyć się programem SAS XML Mapper. W przypadku eksportu mapę należy utworzyć ręcznie. Składnia mapy xml jest opisana w dokumencie SAS® 9.4 XML Libname Engine w rozdziale XMLMap File Syntax. Poniższy fragment pokazuje przykładową mapę XML, proces eksportu oraz wynikowy plik xml.

```
filename mapfile temp;
data _null_;
    file mapfile encoding="utf-8";
    put  '<?xml version="1.0" encoding="UTF-8"?>';
    put  '<SXLEMAP name="class_sample" version="2.1">';
    put  '  <OUTPUT>';
    put  '    <TABLEREF name="class"/>';
    put  '  </OUTPUT>';
    put  '  <TABLE name="CLASS">';
    put  '      <TABLE-PATH syntax="XPath">/DATA/CLASS</TABLE-PATH>';
    put  '      <COLUMN name="Name">';
    put  '          <PATH syntax="XPath">/DATA/CLASS/@Name</PATH>';
    put  '          <TYPE>character</TYPE>';
    put  '          <DATATYPE>string</DATATYPE>';
    put  '          <LENGTH>8</LENGTH>';
    put  '      </COLUMN>';
    put  '  </TABLE>';
    put  '</SXLEMAP>';
run;

libname xmllib xmlv2 "C:\SAS\classv2.xml" xmlmap=mapfile;

data xmllib.class;
    set sashelp.class;
run;
```

Rezultat:

```
<?xml version="1.0" encoding="windows-1250" ?>
<DATA>
<CLASS Name="Alfred" />
<CLASS Name="Alice" />
<CLASS Name="Barbara" />
<CLASS Name="Carol" />
<CLASS Name="Henry" />
<CLASS Name="James" />
<CLASS Name="Jane" />
<CLASS Name="Janet" />
<CLASS Name="Jeffrey" />
<CLASS Name="John" />
<CLASS Name="Joyce" />
<CLASS Name="Judy" />
<CLASS Name="Louise" />
<CLASS Name="Mary" />
<CLASS Name="Philip" />
<CLASS Name="Robert" />
<CLASS Name="Ronald" />
<CLASS Name="Thomas" />
<CLASS Name="William" />
</DATA>
```

Polskie znaki

Obsługa polskich znaków wymaga aby SAS działał w kodowaniu obsługującym polskie znaki (np. latin2 lub wlatin2) oraz wskazania odpowiedniego kodowania znaków w pliku XML (opcja ENCODING= w instrukcji LIBNAME).

4 PC Files (MS Excel, MS Access)

SAS/ACCESS Interface to PC Files udostępnia możliwość interakcji z plikami utworzonymi w programach Microsoft Excel oraz Microsoft Access. Oprócz tego, moduł dodaje również obsługę plików używanych przez Lotus 1-2-3, dBase, JMP, SPSS, Stata oraz Paradox. W tym rozdziale skupimy się na obsłudze formatów Microsoft Excel oraz Access.

Wymagania

Dostęp do danych przechowywanych w arkuszach Excel bądź bazach danych Access z poziomu kodu SAS wymaga posiadania modułu ***SAS/ACCESS Interface to PC Files***. W pewnych przypadkach konieczna jest instalacja komponentu SAS PC Files Server.

Konfiguracja

W ramach modułu ***SAS/ACCESS Interface to PC Files*** otrzymujemy dostęp do szeregu metod, różniących się wymaganiami związanymi z konfiguracją środowiska. Metody te pod tym względem można podzielić na następujące grupy:

- Nie wymagające dalszej konfiguracji:
  - PROC IMPORT oraz PROC EXPORT z DBMS=XLS
  - PROC IMPORT oraz PROC EXPORT z DBMS=XLSX
- Wymagające pakietu Office lub Microsoft Access Database Engine:
  - PROC IMPORT oraz PROC EXPORT z DBMS=EXCEL
  - PROC IMPORT oraz PROC EXPORT z DBMS=ACCESS
  - LIBNAME EXCEL
  - LIBNAME ACCESS
- Wymagające komponentu SAS PC Files Server:
  - LIBNAME PCFILES
  - PROC IMPORT oraz PROC EXPORT z DBMS=EXCELCS
  - PROC IMPORT oraz PROC EXPORT z DBMS=ACCESSCS

Użycie

SAS oferuje następujące metody dostępu do plików MS Excel i MS Access:

- LIBNAME PCFILES – pozwala na korzystanie ze skoroszytu lub bazy danych jak z biblioteki
- LIBNAME EXCEL – pozwala na korzystanie ze skoroszytu jak z biblioteki
- LIBNAME ACCESS – pozwala na korzystania z pliku bazy danych jak z biblioteki
- PROC IMPORT i PROC EXPORT – odczyt i zapis danych z pojedynczego arkusza
  - DBMS = XLS – obsługuje pliki w formacie xls (Excel 97-2003)
  - DBMS = XLSX – obsługuje pliki w formacie xlsx (Excel 2007 i nowsze)
  - DBMS = EXCEL -
  - DBMS = EXCELCS
  - DBMS = ACCESS
  - DBMS = ACCESSCS

Przykładowe użycie:

```
/*PROC EXPORT*/
proc export outfile='C:\sashelp.xlsx'
       data=sashelp.cars
       dbms=xlsx;
       sheet=cars;
run;

/*PROC IMPORT*/
proc import datafile='C:\sashelp.xlsx'
       out=work.cars
       dbms=xlsx;
       sheet=cars;
run;

/*LIBNAME EXCEL*/
libname exceltst excel 'C:\sashelp.xlsx';

data exceltst.class;
       set sashelp.class;
run;

data work.class;
       set exceltst.class;
run;
```

Szczegółowe informacje na temat działania poszczególnych metod oraz przykładowe kody znajdują się na stronie FAQ SAS/ACCESS Interface to PC Files oraz w dokumentacji.

Polskie znaki

Polskie znaki są wspierane przez większość dostępnych metod. Wyjątkiem są PROC IMPORT oraz PROC EXPORT z DBMS=XLS, które nie obsługują polskich znaków.

5 Oracle

Wymagania

Dostęp do danych przechowywanych w bazie danych Oracle z poziomu SAS wymaga posiadania modułu ***SAS/ACCESS Interface to Oracle***. Dodatkowo na komputerze z SAS musi być zainstalowany klient Oracle w odpowiedniej wersji.

Wspierane wersje klientów Oracle to:

- SAS 9.2 – Oracle 9i Release 2
- SAS 9.3 – Oracle 10g
- SAS 9.4 – Oracle 11gR2, Oracle 12c*, Oracle 18c*

  * Do poprawnego działania mogą być wymagane dodatkowe kroki.

Szczegółowe informacje na temat wymagań znajdują się w dokumencie System Requirements dla posiadanej wersji SAS i systemu operacyjnego.

UWAGA. Powyższe wersje dotyczą wsparcia przez oprogramowanie SAS Foundation. Niektóre rozwiązania (np. SAS Customer Intelligence) mogą posiadać własne wymagania dotyczące wersji serwera oraz klienta bazy danych.

Konfiguracja

Oprócz instalacji samego klienta Oracle, SAS wymaga zdefiniowania odpowiednich zmiennych środowiskowych w systemie operacyjnym. Wymagane zmienne to:

- **ORACLE_HOME** – wskazuje na katalog instalacji klienta Oracle
- **ORACLE_SID** – zawiera nazwę domyślnej bazy danych
- **NLS_LANG** – definiuje kodowanie znaków używane przez klienta
- **PATH** (WINDOWS) / **LIBPATH** (AIX) / **SHLIB_PATH** (HP_UX) / **LD_LIBRARY_PATH** (Linux, Solaris) – wskazuje lokalizację folderu **lib** dla klienta Oracle.

W przypadku błędnej konfiguracji zmiennych środowiskowych próba połączenia do bazy może zwrócić następujące komunikaty:
ERROR: The SAS/ACCESS Interface to ORACLE cannot be loaded.
ERROR: Image SASORA found but not loadable.

Polskie znaki

Aby SAS prawidłowo obsługiwał polskie znaki przy imporcie i eksporcie danych z Oracle, wymagane jest spełnienie następujących warunków:

- Kodowanie sesji SAS musi obsługiwać polskie znaki (dostępne kodowania to wlatin2, latin2 oraz utf-8)
- Kodowanie klienta Oracle musi odpowiadać kodowaniu SAS-a.

Kodowanie klienta Oracle ustawia się za pomocą zmiennej środowiskowej NLS_LANG. Poniższa tabelka zawiera zalecane wartości tej zmiennej dla poszczególnych formatów kodowania w SAS.

KODOWANIE SAS	NLS_LANG
LATIN2	POLISH_POLAND.EE8ISO8859P2
WLATIN2	POLISH_POLAND.EE8MSWIN1250
UTF-8	POLISH_POLAND.AL32UTF8

6 ODBC

Wymagania

Dostęp do danych przechowywanych w bazie danych za pomocą mechanizmu ODBC z poziomu SAS wymaga posiadania modułu ***SAS/ACCESS Interface to ODBC***. Metoda ta pozwala na dostęp do praktycznie dowolnej bazy danych (m.in. Oracle, SQL Server, MySQL, PostgreSQL). Należy mieć jednak na uwadze, że wydajność takiego połączenia może być niższa niż w przypadku korzystania z modułu SAS/ACCESS dedykowanego dla danej bazy.

Do poprawnego działania modułu SAS/ACCESS Interface to ODBC wymagane są:

- Instalacja administratora ODBC (nie dotyczy systemów Windows)
- Instalacja sterowników ODBC docelowych baz danych (np. Oracle czy SQL Server)
- Definicja źródeł danych (DSN)

Konfiguracja

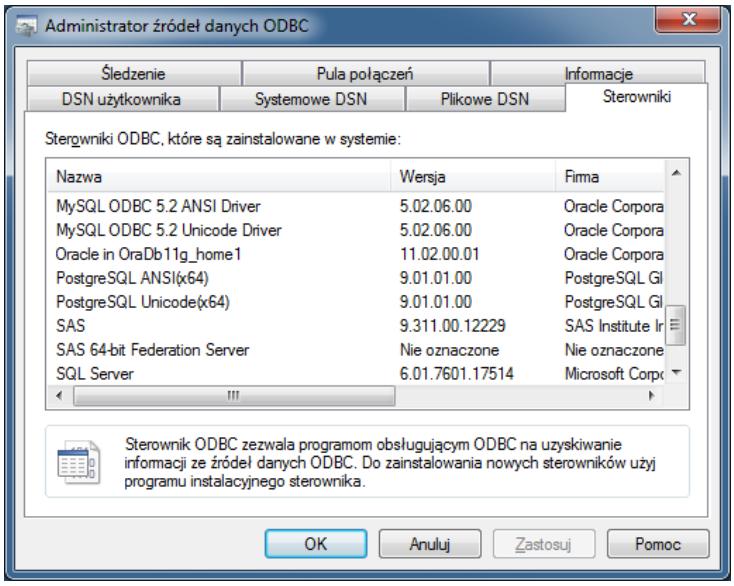
Dokładny przebieg tego procesu różni się w zależności od systemu operacyjnego, na którym zainstalowany jest SAS.

UWAGA. Oprócz konfiguracji samego ODBC niektóre bazy danych wymagają dodatkowo konfiguracji aplikacji klienckiej. W przypadku problemów z działaniem połączenia należy wykonać instrukcje wymienione w rozdziale dla wybranej bazy danych.

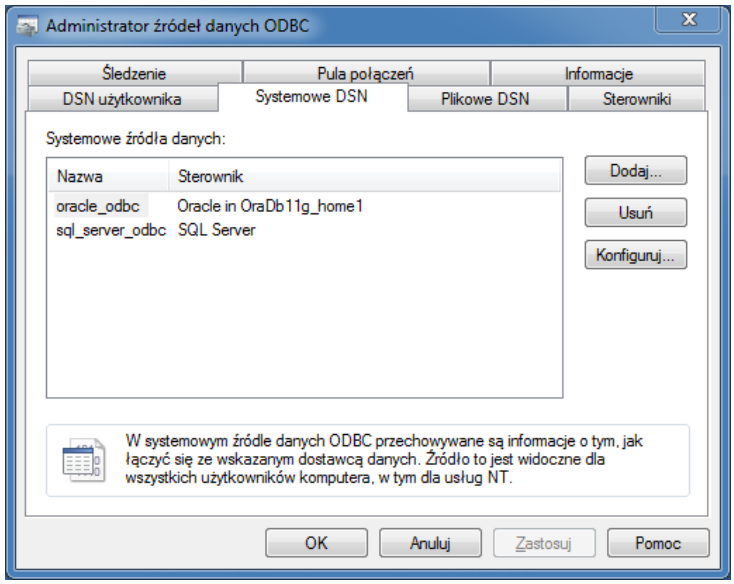
Windows

Na systemie Windows administrator ODBC jest wbudowany w system. Aby się do niego dostać należy wejść w *Panel sterowania -> Narzędzia administracyjne -> Źródła danych (ODBC).*

W zakładce *Sterowniki* znajduje się lista zainstalowanych sterowników ODBC. Aby dodać nowe sterowniki należy pobrać je ze strony producenta i zainstalować. Bitowość sterowników musi być zgodna z bitowością SAS.



Po zainstalowaniu sterownika w zakładce *Systemowe DSN* należy zdefiniować połączenie do źródła danych. Proces definicji połączenia różni się w zależności od używanego sterownika. Po zdefiniowaniu połączenia powinno pojawić się ono na liście systemowych źródeł danych.



UNIX

Administrator ODBC

W przypadku systemów z rodziny UNIX, administrator ODBC jest samodzielną aplikacją i często nie jest domyślnie zainstalowany. SAS jest w stanie współpracować z dowolnym administratorem ODBC, jednak zaleca się korzystanie z **unixODBC**. Program ten jest dostępny w większości standardowych repozytoriów pakietów. Instalacja w tym przypadku wymaga praw root-a oraz (najczęściej) dostępu do Internetu. Instalacja pakietu to w większości przypadków wykonanie pojedynczej komendy, np.

```
yum install unixODBC.x86_64
```

W wyniku instalacji w katalogu /etc powinny zostać utworzone pliki ***odbc.ini*** oraz **odbcinst.ini**.

Sterownik ODBC do bazy danych

Po zainstalowaniu administratora ODBC należy pobrać i zainstalować sterowniki ODBC. Sposób pobierania i instalacji zależy tutaj od konkretnej bazy danych. Sterowniki niektórych baz danych (np. MySQL oraz PortgreSQL) są dostępne bezpośrednio w standardowych repozytoriach pakietów. Pobieranie i instalacja w tym przypadku sprowadza się do wykonanie jednego polecenia, np.

```
yum install postgresql-odbc.x86_64
yum install mysql-connector-odbc.x86_64
```

W przypadku innych baz danych szczegółowych informacji na temat pobierania i instalacji należy szukać na stronie producenta sterownika.

Po zakończeniu instalacji należy otworzyć plik /etc/odbcinst.ini i dodać wpis dla nowego sterownika (niektóre sterowniki robią to automatycznie podczas instalacji).

Jeżeli korzystamy z unixODBC, zainstalowane sterowniki ODBC można sprawdzić wykonując polecenie "odbcinst -q -d"

Przykładowy wpis w pliku odbcinst.ini:

```
# Driver from the postgresql-odbc package
# Setup from the unixODBC package
[PostgreSQL]
Description   = ODBC for PostgreSQL
Driver        = /usr/lib/psqlodbcw.so
Setup         = /usr/lib/libodbcpsqlS.so
Driver64      = /usr/lib64/psqlodbcw.so
Setup64             = /usr/lib64/libodbcpsqlS.so
FileUsage     = 1
```

Źródło danych

Po zainstalowaniu sterownika ODBC należy zdefiniować źródło danych (DSN). Definicję definiuje się ręcznie w pliku /etc/odbc.ini. Zawiera ona definicję połączenia do konkretnej bazy danych. Ustawiane są tu m.in. następujące parametry:

- Sterownik używany do połączenia
- Nazwa lub adres IP serwera bazy danych
- Port nasłuchu serwera bazy danych
- Nazwa schematu
- Parametry połączenia (kodowanie znaków itp.)

Dokładny zestaw parametrów zależy od sterownika.

W przypadku unixODBC zdefiniowane źródła danych można sprawdzić wykonując polecenie "odbcinst -q -s".

Przykładowa definicja źródła danych:

```
[PSQL_TEST]
Driver          = PostgreSQL
Server          = localhost
Port            = 9432
Database        = SharedServices
```

Po utworzeniu definicji źródła danych, należy zweryfikować poprawność definicji przy użyciu zewnętrznego narzędzia, np. **isql**. Przykładowe polecenie to:

```
isql -v -n <DSN> <LOGIN> <HASŁO>
```

Zmienne środowiskowe

SAS/ACCESS Interface to ODBC do poprawnego działania wymaga następujących zmiennych środowiskowych:

- **PATH** – zmienna powinna zawierać katalogu **lib** administratora ODBC
- **ODBCSYSINI** – zmienna wskazująca folder zawierający pliki z definicjami serwerów (odbcinst.ini) oraz źródeł danych (odbc.ini). Zazwyczaj jest to katalog **/etc**.
- **LIBPATH** (AIX) / **SHLIB_PATH** (HP_UX) / **LD_LIBRARY_PATH** (Linux, Solaris) – zmienna powinna zawierać wskazania na katalogi zawierające pliki sterowników ODBC, wskazane w pliku odbcinst.ini.

Polskie znaki

Aby polskie znaki były wyświetlane poprawnie, konieczne jest spełnienie poniższych warunków:

- Sterownik ODBC musi wspierać polskie znaki
- Źródło danych musi używać odpowiedniego kodowania znaków
- Kodowanie sesji SAS musi wspierać polskie znaki

W przypadku problemów z obsługą polskich znaków podczas korzystania z unixODBC, rozwiązaniem często jest ustawienie poniższej zmiennej środowiskowej:

- **EASYSOFT_UNICODE=YES**

UWAGA. Niektóre bazy danych wymagają różnych sterowników w zależności od wybranego kodowania znaków. Najczęściej dotyczy to przesyłania danych w formacie Unicode (UTF-8). Szczegółowych informacji należy szukać na stronie producenta bazy.

7 SQL Server

Wymagania

Dostęp do danych przechowywanych w bazie danych SQL Server z poziomu SAS wymaga posiadania modułu ***SAS/ACCESS Interface to Microsoft SQL Server***.

Dodatkowo konieczna jest konfiguracja źródła danych w oparciu o dostarczony sterownik.

Wspierane wersje Microsoft SQL Server to:

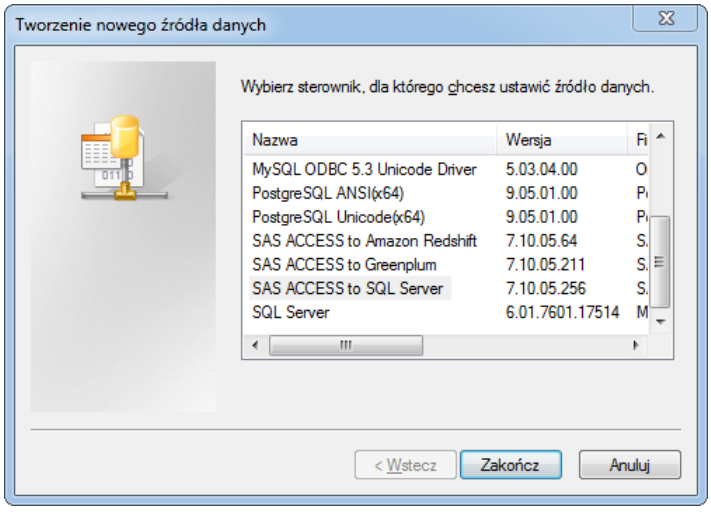
- SAS 9.2 – SQL Server 7 lub nowszy
- SAS 9.3 – SQL Server 7 lub nowszy
- SAS 9.4 (M2 i starsze) – SQL Server 2008 lub nowszy
- SAS 9.4 (M3 i nowsze) – SQL Server 2012 lub nowszy

Konfiguracja

Moduł SAS/ACCESS Interface to Microsoft SQL Server wykorzystuje technologię ODBC. W odróżnieniu od modułu SAS/ACCESS Interface to ODBC, wszystkie wymagane komponenty zewnętrzne (zarządca ODBC oraz sterowniki ODBC) są dostarczane razem z modułem.

Windows

W przypadku systemów Windows konfiguracja modułu SAS/ACCESS Interface to Microsoft SQL Server ogranicza się do utworzenia źródła danych ODBC. Jako sterownik należy użyć **SAS ACCESS to SQL Server**.



UNIX

Od wersji SAS 9.4M3 sterowniki oraz zarządca ODBC są instalowane automatycznie w katalogu **<SAS_HOME>/AccessClients/9.4/SQLServer**. We wcześniejszych wersjach w procesie instalacji była możliwość zmiany domyślnej lokalizacji.

```
Przez <SAS_HOME> rozumiemy folder, w którym SAS został zainstalowany, np.
/opt/sas/sashome. W przykładach podanych poniżej wartość tę należy zastąpić właściwą
ścieżką do folderu.
```

W katalogu instalacji sterowników umieszczone są wstępnie wypełnione pliki **odbc.ini** oraz **odbcinst.ini**. Do ich poprawnego działania wymagane są następujące zmiany:

- W obu plikach należy usunąć fragmenty "<install_path>" i w ich miejsce wstawić właściwą ścieżkę do folderu, w którym pliki się znajdują (np. "**<SAS_HOME>/AccessClients/9.4/SQLServer**").
- W pliku **odbc.ini** należy poprawić wartości parametrów **Address**, **Database**, **HostName** oraz **PortNumber.**

  Oprócz modyfikacji wymienionych plików konieczne jest ustawienie jeszcze następujących zmiennych środowiskowych:
- **LIBPATH** (AIX) / **SHLIB_PATH** (HP_UX) / **LD_LIBRARY_PATH** (Linux, Solaris) – do zmiennej należy dodać ścieżkę do katalogu **<SAS_HOME>/AccessClients/9.4/SQLServer/lib**
- **ODBCINI** – zmienna ta powinna zawierać ścieżkę do pliku odbc.ini (np. "**<SAS_HOME>/AccessClients/9.4/SQLServer/odbc.ini**")

Polskie znaki

Obsługa polskich znaków wymaga ustawienia odpowiedniego kodowania sesji w połączeniu ODBC. Kodowanie to ustawia się przy użyciu IANAAPPCODEPAGE w definicji źródła danych (DSN). Wartość opcji musi odpowiadać kodowaniu sesji SAS. Dostępne wartości opcji IANAAPPCODEPAGE dla języka polskiego zostały wymienione w poniższej tabelce:

ENCODING	Opcja w odbc.ini
LATIN2 (ISO-8859-2)	IANAAPPCODEPAGE=5
WLATIN2 (WINDOWS_1252)	IANAAPPCODEPAGE=1252
UTF-8	IANAAPPCODEPAGE=106

Dodatkowo należy pamiętać, że dane tekstowe na SQL Server są przechowywane przy użyciu różnych typów kolumn, w zależności od używanego kodowania znaków. W przypadku tekstów w kodowaniu innym niż UNICODE, wykorzystywane są kolumny typu CHAR, VARCHAR i TEXT. W przypadku tekstów w kodowaniu UNICODE są to natomiast NCHAR, NVARCHAR i NTEXT. Więcej szczegółów można znaleźć w nocie 36652.

8 MySQL

Wymagania

Dostęp do danych przechowywanych w bazie danych MySQL z poziomu SAS wymaga posiadania modułu ***SAS/ACCESS Interface to MySQL***. Dodatkowo na komputerze z SAS musi być zainstalowany klient MySQL w odpowiedniej wersji.

Minimalne wspierane wersje klientów to:

- SAS 9.2 – MySQL Client 4.1 lub 5.0
- SAS 9.3 – MySQL Client 5.1 lub 5.5
- SAS 9.4 (M3 i starsze) – MySQL Client 5.1 lub nowszy
- SAS 9.4 (M4 i nowsze) – MySQL Client 5.6 lub nowszy

Bitowość klienta MySQL musi być zgodna z bitowością SAS.

Szczegółowe informacje na temat wymagań znajdują się w dokumencie System Requirements dla posiadanej wersji SAS i systemu operacyjnego.

```
Od wersji 9.4 moduł SAS/ACCESS Interface to MySQL nie jest dostępny na platformie HP-UX.
```

Konfiguracja

Oprócz instalacji klienta bazy danych, do poprawnego działania modułu SAS/ACCESS Interface to MySQL wymagane jest ustawienie odpowiednich zmiennych środowiskowych.

Wymagane zmienne to:

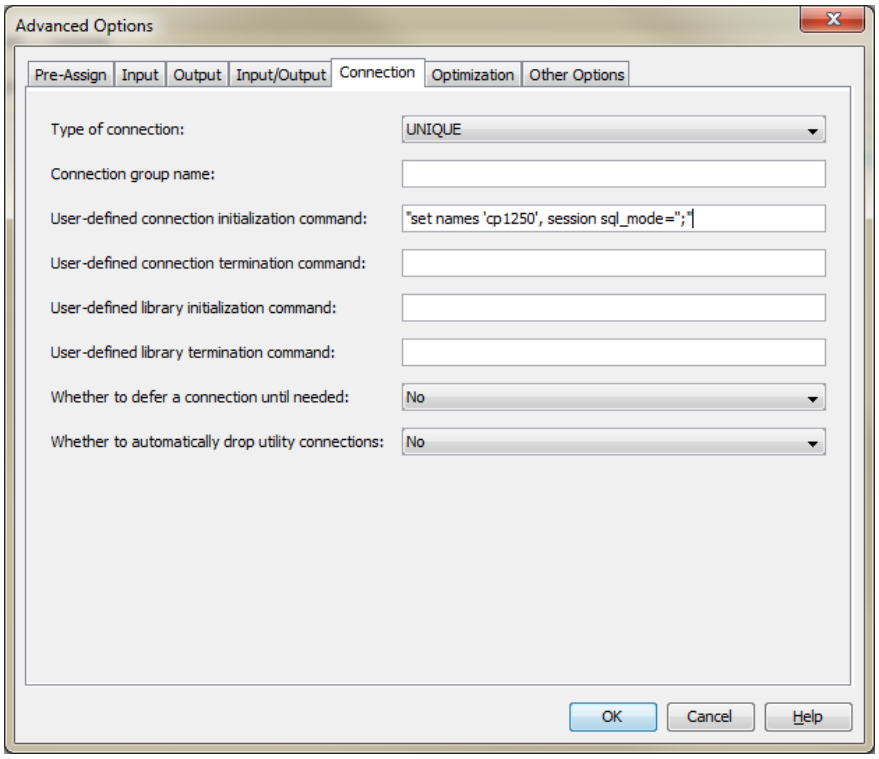
- **PATH** (WINDOWS) – do zmiennej tej należy dodać katalog zawierający plik libmysql.dll. Zazwyczaj jest to katalog **C:\mysql\bin** bądź **C:\Program Files\mysql\mysql server 5.1\bin**.
- **LIBPATH** (AIX) / **SHLIB_PATH** (HP_UX) / **LD_LIBRARY_PATH** (Linux, Solaris) – do zmiennej tej należy dodać katalog zawierający plik libmysqlclient.so. Domyślnie jest to katalogu **/usr/lib64/mysql**.

Polskie znaki

Kodowanie znaków dla biblioteki MySQL ustawiania się na poziomie instrukcji **libname**, w ramach opcji **dbconinit**. Wartość, jaką należy ustawić zależy od wartości zmiennej **encoding** dla sesji SAS. Odpowiednie wartości zostały przedstawione w poniższej tabeli:

ENCODING	DBCONINIT
LATIN2	set names 'iso8859-2'
WLATIN2	set names 'cp 1250'
UTF-8	set names 'utf8'

W przypadku definicji biblioteki w metadanych, opcję tę ustawia się w sekcji *Advanced Options* w zakładce *Connection* w polu *User-defined connection initialization command*. Przykładowa konfiguracja przedstawiona została na poniższym zrzucie ekranu.


Advanced Options
Pre-Assign
Input
Output
Input/Output
Connection
Optimization
Other Options
Type of connection:
UNIQUE
Connection group name:
User-defined connection initialization command:
"set names 'cp1250', session sql_mode='';"
User-defined connection termination command:
User-defined library initialization command:
User-defined library termination command:
Whether to defer a connection until needed:
No
Whether to automatically drop utility connections:
No
OK
Cancel
Help

9 PostgreSQL

Wymagania

Dostęp do danych przechowywanych w bazie danych PostgreSQL z poziomu SAS wymaga posiadania modułu ***SAS/ACCESS Interface to PostgreSQL***. Moduł ten jest dostępny od SAS 9.4

Wspierane wersje PostgreSQL to:

- SAS 9.4 (M3 i starsze) – PostgreSQL 9.1.9 lub nowszy
- SAS 9.4 (M4 i M5) – PostgreSQL 9.1.23 lub nowszy
- SAS 9.4 (M6) – PostgreSQL 9.5.14 lub nowszy

Konfiguracja

SAS/ACCESS Interface to PostgreSQL pozwala na dwa typy połączenia:

- Połączenie bezpośrednie – w instrukcji LIBNAME podajemy jawnie serwer, port, bazę danych, użytkownika oraz hasło
- Połączenie pośrednie (ODBC) – w instrukcji LIBNAME podajemy nazwę źródła danych ODBC, do którego chcemy się podłączyć.

Typy te obrazuje poniższy fragment kodu:

```
/* Połączenie bezpośrednie */
libname psql_dir postgres server="psql" port=5432 database="SAS" user="sas" pw="#####";

/* Połączenie pośrednie (ODBC) */
libname psql_dsn postgres dsn="psql_odbc" user="sas" pw="#####";
```

Połączenie bezpośrednie nie wymaga żadnych dodatkowych kroków konfiguracyjnych. Połączenie pośrednie wymaga zdefiniowania źródła danych ODBC. Szczegółowe informacje na temat konfigurowania źródła danych ODBC znajdują się w rozdziale ODBC.

Polskie znaki

Aby SAS prawidłowo obsługiwał polskie znaki przy imporcie i eksporcie danych z PostgreSQL, wymagane jest spełnienie następujących warunków:

- Kodowanie sesji SAS musi obsługiwać polskie znaki (dostępne kodowania to wlatin2, latin2 oraz utf-8)
- Kodowanie klienta PostgreSQL musi odpowiadać kodowaniu SAS-a.

Windows

W przypadku systemów Windows kodowanie znaków dla biblioteki PostgreSQL ustawiania się na poziomie instrukcji **libname**, w ramach opcji **dbconinit**. Wartość, jaką należy ustawić zależy od wartości zmiennej **encoding** dla sesji SAS. Odpowiednie wartości zostały przedstawione w poniższej tabeli:

ENCODING	DBCONINIT
LATIN2	set client_encoding to 'latin2'
WLATIN2	set client_encoding to 'cp 1250'
UTF-8	set client_encoding to 'utf8'

UNIX

W przypadku systemów Linux / UNIX i AIX kodowanie klienta ustawia się za pomocą zmiennej środowiskowej PGCLIENTENCODING. Poniższa tabelka zawiera zalecane wartości tej zmiennej dla poszczególnych formatów kodowania w SAS.

KODOWANIE SAS	NLS_LANG
LATIN2	LAIN2
WLATIN2	WIN1250
UTF-8	UTF8

10 Teradata

Wymagania

Wspierane wersje klientów Teradata to:

- SAS 9.2 – TTU 8.2 i nowsze
- SAS 9.3 – TTU 12 i nowsze
- SAS 9.4 (M2 i starsze) – TTU 13.10 i nowsze
- SAS 9.4 (M3) – TTU 14.10 i nowsze
- SAS 9.4 (M4 i nowsze) – TTU 15.10 i nowsze

Konfiguracja

Teradata, oprócz podstawowego klienta bazy danych, dostarcza szereg narzędzi pozwalających na przyspieszenie operacji ładowania i eksportu danych. W nowszych wersjach służy do tego mechanizm Teradata Parallel Transporter (TPT), natomiast wcześniej funkcjonalność ta była rozbita pomiędzy dwa narzędzia: FastExport oraz MultiLoad. SAS potrafi używać każdego z wymienionych komponentów, jednakże wymaga to odpowiedniej konfiguracji.

Szczegółowe informacje na temat konfiguracji SAS i Teradaty można znaleźć w dokumencie SAS/ACCESS Interface to Teradata white paper.

Klient

- Windows – nie ma potrzeby wykonywania dodatkowych kroków. Należy jedynie upewnić się, że klient jest w stanie połączyć się z bazą. Można to sprawdzić np. używając narzędzia Teradata BTEQ
- UNIX – SAS wymaga, aby katalog instalacji klienta (zazwyczaj jest to ***/opt/teradata/client***) został dodany do zmiennej środowiskowej **LIBPATH** (AIX) / **SHLIB_PATH** (HP-UX) / **LD_LIBRARY_PATH** (Linux, Solaris).

Teradata Parallel Transporter (TPT)

Obecnie zalecany mechanizm do ładowania i eksportowania danych z Teradaty.

- WINDOWS
  - Komponent TPT API musi być zainstalowany na serwerze SAS.
  - Do zmiennej środowiskowej **PATH** należy dodać biblioteki tego modułu.
  - Do zmiennej środowiskowej **TKPATHX8664** należy dopisać katalog zawierający plik **sasiotpt.dll** (domyślnie ***C:\Program Files\SASHome\SASFoundation\9.4\access\sasexe***). Jeżeli zmienna taka nie istnieje, należy ją utworzyć.
- UNIX
  - Do zmiennej środowiskowej **LIBPATH** (AIX) / **SHLIB_PATH** (HP_UX) / **LD_LIBRARY_PATH** (Linux, Solaris) należy dodać katalogi z bibliotekami TPT. Domyślnie są to ***/opt/teradata/client/14.00/tbuild/lib64*** oraz ***/opt/teradata/client/14.00/lib64***
  - Do zmiennej środowiskowej **NLSPATH** należy dodać katalog msg klienta. Domyślnie jest to ***/opt/teradata/client/14.00/tbuild/msg64***.

UWAGA. W przypadku wpisów w zmiennej środowiskowej PATH (Windows) oraz LIBPATH (AIX) / SHLIB_PATH (HP_UX) / LD_LIBRARY_PATH (Linux, Solaris) należy upewnić się, że zmienne te zawierają jedynie ścieżki do katalogów z 64-bitowymi wersjami komponentu TPT, bądź katalogi do wersji 64-bitowych występują przed katalogami do wersji 32-bitowych.

Aby zweryfikować poprawność działania mechanizmu TPT z poziomu SAS można wykonać poniższy kod:

```
Libname x teradata server=mydbc user=dtest pw=xxxxx ;
Data x.new (fastload=yes) ; x=99; run;
```

Jeżeli wszystko działa poprawnie, wygenerowany log powinien wyglądać następująco:

```
NOTE: The data set X.new has 1 observations and 1 variables
NOTE: Teradata connection: TPT Fastload has inserted 1 row(s).
NOTE: Data statement used (Total process time):
       real time 4.39
       cpu time 0.29
```

FastExport

- WINDOWS
  - Narzędzie Teradata FastExport Utility musi być zainstalowane na serwerze SAS.
  - Do zmiennej środowiskowej **PATH** należy dodać katalog zawierający plik **fexp.exe** (domyślnie ***C:\Program Files\Teradata\Client\13.0\bin***) oraz katalog zawierający plik **sasaxsm.dll** (domyślnie ***C:\Program Files\SASHome\SASFoundation\9.4\access\sasexe***).
- UNIX
  - Do zmiennej środowiskowej **LIBPATH** (AIX) / **SHLIB_PATH** (HP_UX) / **LD_LIBRARY_PATH** (Linux, Solaris) należy dodać katalog zawierający plik sasaxsm.so (domyślnie ***/sas/SASHome/SASFoundation/9.4/sasexe***).
  - Do zmiennej środowiskowej **PATH** należy dodać katalog zawierający program Teradata FastExport Utility (fexp). Domyślnie jest to katalog ***/usr/bin***.

MultiLoad

- WINDOWS
  - Narzędzie Teradata MultiLoad Utility musi być zainstalowane na serwerze SAS.
  - Do zmiennej środowiskowej **PATH** należy dodać katalog zawierający plik **mload.exe** (domyślnie ***C:\Program Files\Teradata\Client\13.0\bin***) oraz katalog zawierający plik **sasmlam.dll** oraz **sasmlne.dll** (domyślnie ***C:\Program Files\SASHome\SASFoundation\9.4\access\sasexe***).
- UNIX
  - Do zmiennej środowiskowej **LIBPATH** (AIX) / **SHLIB_PATH** (HP_UX) / **LD_LIBRARY_PATH** (Linux, Solaris) należy dodać katalog zawierający pliki sasmlam.so oraz sasmlne.so (domyślnie ***/sas/SASHome/SASFoundation/9.4/sasexe***).
  - Do zmiennej środowiskowej **PATH** należy dodać katalog zawierający program Teradata MultiLoad Utility (mload). Domyślnie jest to katalog ***/usr/bin***.

Polskie znaki

Informacje o stronie kodowej, w jakiej będzie się odbywać wymiana danych pomiędzy klientem Teradaty a SAS-em ustawia się w pliku konfiguracyjnym klienta Teradaty ***clispb.dat***. Aby klient Teradaty prawidłowo obsługiwał polskie znaki, we wskazanym pliku należy ustawiać następujące opcje:

```
charset_type=N
charset_id=LATIN2_0X | LATIN1250_1A0 | UTF8
```

Opcja charset_type wskazuje, czy opcja charset_id będzie zawierać nazwę (charset_type=N) czy też identyfikator (charset_type=C). Opcja charset_id jest ustawiana w zależności od platformy i kodowania sesji SAS.

Lokalizację pliku ***clispb.dat*** wskazuje zmienna środowiskowa COPLIB, zawierająca katalog, w którym plik jest przechowywany. Plik ten jest tworzony w ramach instalacji klienta Teradaty. Zalecane jest jednak utworzenie kopii zapasowej na wypadek ewentualnych problemów we współpracy między Teradatą a SAS-em wynikających z modyfikacji wprowadzanych w pliku. Zmienną COPLIB dla SAS-a w systemie UNIX można ustawić m.in. w pliku ***<SASHome>/SASFoundation/9.X/bin/sasevn_local*** lub ***sasv9_local.cfg***. W przypadku systemu Windows opcję tę należy ustawić w pliku konfiguracyjnym ***sasv9.cfg***.

Ustawienie zmiennej można sprawdzić z poziomu SAS-a uruchamiając następujący kod:

```
%PUT %SYSGET(COPLIB);
```

11 Hadoop

Wymagania

W przypadku technologii Hadoop oraz SAS, dostępnych jest wiele metod dostępu. W tym dokumencie przedstawiamy podstawową metodę, czyli ***SAS/ACCESS Interface to Hadoop***. Metoda ta polega na połączeniu się z podsystemem Hive (HiveServer2).

Lista wspieranych wersji oprogramowania Hadoop zależy od kilku czynników, takich jak metoda połączenia, wersja SAS-a oraz dystrybucja Hadoop-a. Szczegółowe zestawienie wspieranych kombinacji znajduje się na stronie Support for Hadoop.

Konfiguracja

W celu konfiguracji połączenia do Hadoop-a należy na początek pobrać pliki z serwera Hadoop. W tym celu należy uruchomić aplikację **SAS Deployment Manager** (np. **<SAS_HOME>/SASDeploymentManager/9.4/sasdm**) a następnie z sekcji **Hadoop Configuration** wybrać **Configure Hadoop Client Files**. Następnie należy wskazać posiadaną dystrybucję W ten sposób zostaną pobrane z Hadoopa pliki, które tworzą klienta Hadoopa.

Proces ten można również wykonać ręcznie, przy pomocy skryptu **hadooptracer.py**. Plik ten znajduje się w folderze **<SAS_HOME>/SASHadoopConfigurationLibraries/2.6/data**. Dostępny jest również do pobrania ze strony http://support.sas.com/kb/63/094.html. Strona ta zawiera również skróconą instrukcję jak użyć tego skryptu do konfiguracji klienta Hadoop-a.

Po uruchomieniu skryptu i stworzeniu przed niego wymaganych folderów należy ja przenieść na komputer z SAS-em (jeżeli SAS jest na innym komputerze niż HiveServer2) i ustawić dla SAS-a zmienne środowiskowe: **SAS_HADOOP_CONFIG_PATH** i **SAS_HADOOP_JAR_PATH**. Te zmienne wskazują odpowiednio na foldery z: jar'ami Hadoopa (które tworzą klienta) i plikami konfiguracyjnymi.

Zmienne dla SAS-a w systemie UNIX można ustawić m.in. w pliku sasv9_local.cfg (a w przypadku systemu Windows sasv9.cfg.), np.:

```
-set SAS_HADOOP_JAR_PATH "/third_party/Hadoop/jars"
```

Można je też ustawić/zmienić ustawienie w ramach sesji w poleceniu:

```
options set=SAS_HADOOP_JAR_PATH="C:\third_party\Hadoop\jars ";
```

Faktyczne ustawienie zmiennych można sprawdzić uruchamiając w SAS-ie następujący kod:

```
%put %quote(%sysget(SAS_HADOOP_JAR_PATH));
%put %quote(%sysget(SAS_HADOOP_CONFIG_PATH));
```

Szczegółowe instrukcje konfiguracji uzależnione są m.in. od wykorzystywanej funkcjonalności SAS-a i komponentów Hadoopa. Pełny zestaw instrukcji znajduje się w dokumencie SAS 9.4 Hadoop Configuration Guide for Base SAS and SAS/ACCESS.

Polskie znaki

Nie zaobserwowaliśmy problemów z obsługą polskich znaków przy korzystaniu z modułu ***SAS/ACCESS Interface to Hadoop***. Polskie znaki powinny działać bez żadnej dodatkowej konfiguracji. Jedynym wymogiem jest, aby SAS działał w kodowaniu (encoding), obsługującym polskie znaki. Kodowania te to:

- latin2
- wlatin2
- utf-8

12 Dodatkowe informacje

Niniejszy dokument zawiera wyciąg najważniejszych informacji niezbędnych do konfiguracji dostępu do zewnętrznych źródeł danych z poziomu SAS.

- System Requirements

	WINDOWS	LINUX	SOLARIS	AIX	HP-UX
SAS 9.2	32-bit | 64-bit	64-bit	32-bit | 64-bit	64-bit	64-bit
SAS 9.3	32-bit | 64-bit	64-bit	32-bit | 64-bit	64-bit	64-bit
SAS 9.4	32-bit | 64-bit	64-bit	32-bit | 64-bit	64-bit	Itanium

- Configuration Guide

	WINDOWS	UNIX
SAS 9.2M3	32-bit | 64-bit	
SAS 9.3	32-bit | 64-bit	64-bit
SAS 9.4	32-bit | 64-bit	64-bit

- SAS/ACCESS for Relational Databases

	LINK
SAS 9.2	HTML | PDF
SAS 9.3	HTML | PDF
SAS 9.4	HTML | PDF