

# **Jetzt aber Prompt Eingabeaufforderungen im Enterprise Guide**

Daniel Schulte  
viadee Unternehmensberatung  
Anton-Bruchhausen-Straße 8  
48147 Münster  
daniel.schulte@viadee.de

## **Zusammenfassung**

Der Enterprise Guide hat in vielen Bereichen den klassischen Displaymanager weitestgehend verdrängt. Um EG Projekte zu dynamisieren oder auch gegen Fehleingaben abzusichern können diese mittels Eingabeaufforderungen erweitert werden. Das Prompting ermöglicht dabei von einfachen Parametern bis hin zu kompletten Masken eine Vielzahl an Möglichkeiten.

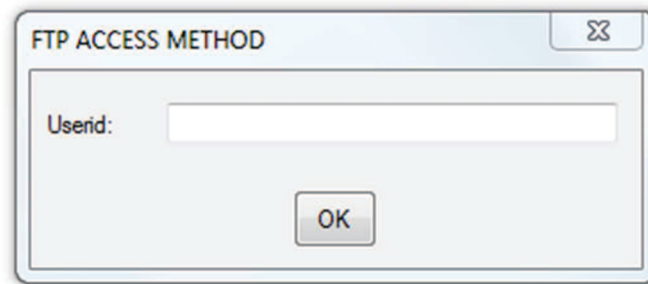
**Schlüsselwörter:** Prompt, Eingabeaufforderungen, SAS Enterprise Guide, AuthDomain, Passwort, Zugangsdaten

## **1 Motivation**

Immer mehr Umgebungen werden von SAS Foundation zum SAS Enterprise Guide (EG) umgestellt. Damit wird zwar eine deutlich modernere Umgebung bereitgestellt, die ein deutlich strukturierteres Arbeiten ermöglicht, aber auch einige Funktionen nicht mehr beinhaltet. Eine davon ist z.B. die Möglichkeit, per PROMPT dem Nutzer Eingaben abzuverlangen. An dieser Stelle kann der Eingabeaufforderungsmanager des EG weiterhelfen. Darüber hinaus gibt es aber auch noch weitere Möglichkeiten z.B. Zugangsdaten einzugeben und die Metadaten zu verlagern.

## **2 Ausgangslage**

In Bereichen, die mit dem SAS Foundation als Gruppe arbeiten, erstellen sich diese häufig Muster und allgemeine Codeblöcke, die allen zur Verfügung gestellt werden. Wenn es nun um den Zugriff auf externe Ressourcen geht, werden in der Regel Benutzername und Passwort benötigt. Um diese eben nicht im Code zu hinterlegen, kann man ein PROMPT nutzen, um dies interaktiv abzufragen:

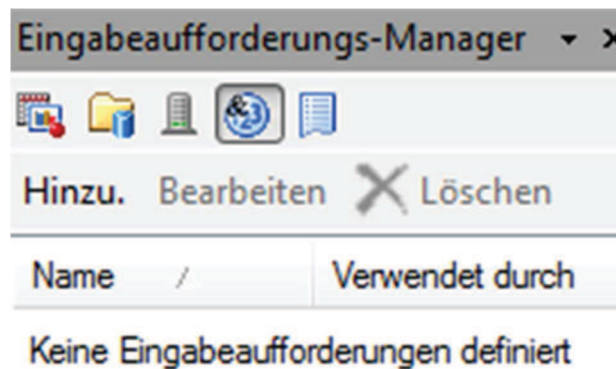


**Abbildung 1:** Prompt in SAS Foundation

Diese Möglichkeit wird jedoch vom Enterprise Guide nicht unterstützt. Es bedarf also anderer Methoden, SAS Code bereitzustellen und Zugangsdaten dynamisch abzufragen.

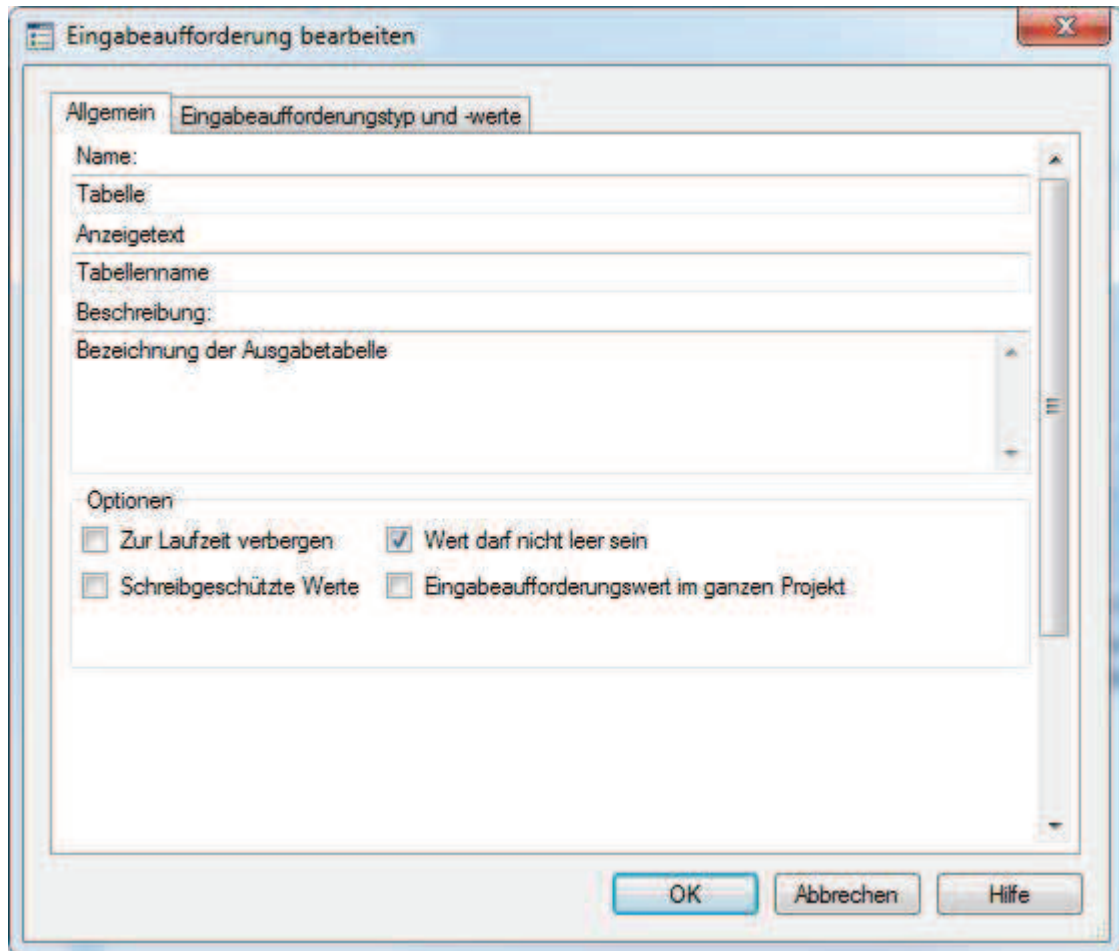
### 3 Eingabeaufforderungs-Manager

Mit umfangreicher konfigurierbaren Eingabeaufforderungen (engl. Prompts) kann man im Enterprise Guide deutlich mehr machen als nur nach Benutzer-Namen oder Passwörtern zu fragen. Diese lassen sich je Programmmodul zusammenstellen und auch datengetrieben füllen.

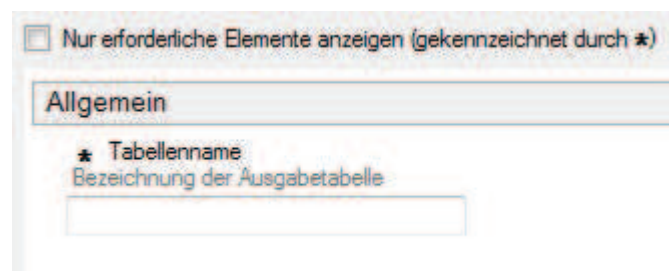


**Abbildung 2:** Eingabeaufforderungs-Manager

Die Eingabeaufforderungen werden je EG Projekt definiert und können dann einzelnen Programmmodulen hinzugefügt werden.



**Abbildung 3:** Einzelne Eingabeaufforderung



**Abbildung 4:** Darstellung der allg. Parameter

Der Name wird als Makrovariable deklariert und bekommt den jeweilig zugewiesenen Wert. Diese Bezeichnung muss daher den Namenskonventionen für Makrovariablen entsprechen. Je nach Typ wird jedoch nicht nur eine Makrovariable erzeugt und der Name darf dann auch ggf. nicht alle 32 Zeichen belegen. Anzeigetext und Beschreibung können mit freiem Text belegt werden.

### 3.1 Typen von Eingabeaufforderungen

Vom Typ her können unterschiedliche Arten von Eingabeaufforderungen definiert werden. Hierzu gehören unter anderem:

**Tabelle 1:** Typen von Eingabeaufforderungen

Typ	Beschreibung	Macros
Text	Einfache Zeichenkette	Prompt_Name Wenn mehr als eine Eingabe möglich ist:: Prompt_Name_Count Prompt_name1 Prompt_name2 .... Prompt_nameN
Nummerisch	Erfassung von Zahlenwerten. Hier sind lediglich integer / ganzzahlige Werte möglich	Prompt_Name Wenn mehr als eine Eingabe möglich ist:: Prompt_Name_Count Prompt_name1 Prompt_name2 .... Prompt_nameN
Nummerischer Bereich	Eingabe eines Zahlenbereiches. Hier werden ebenfalls ganzzahlige Werte erwartet.	Prompt_name_min Prompt_name_max
Datum	Datumswerte können valide erfasst werden. Darüber hinaus gibt es auch relative Werte wie „gestern“ oder „vor N Tagen“	Prompt_Name Datum im date9. Format (01Mar2018)
Datumsbereich	Analog zum Datum können hier Bereiche ausgewählt und auch relativ definiert werden.	Prompt_Name_min Prompt_Name_max Datum im date9. Format (01Apr2018)
Datei / Directory	Dateien / Pfade können so interaktiv belegt werden	Prompt_Name Prompt_name_server

Eine Übersicht der jeweilig verfügbaren Makrovariablen kann über das Kontextmenü des Managers eingesehen werden:

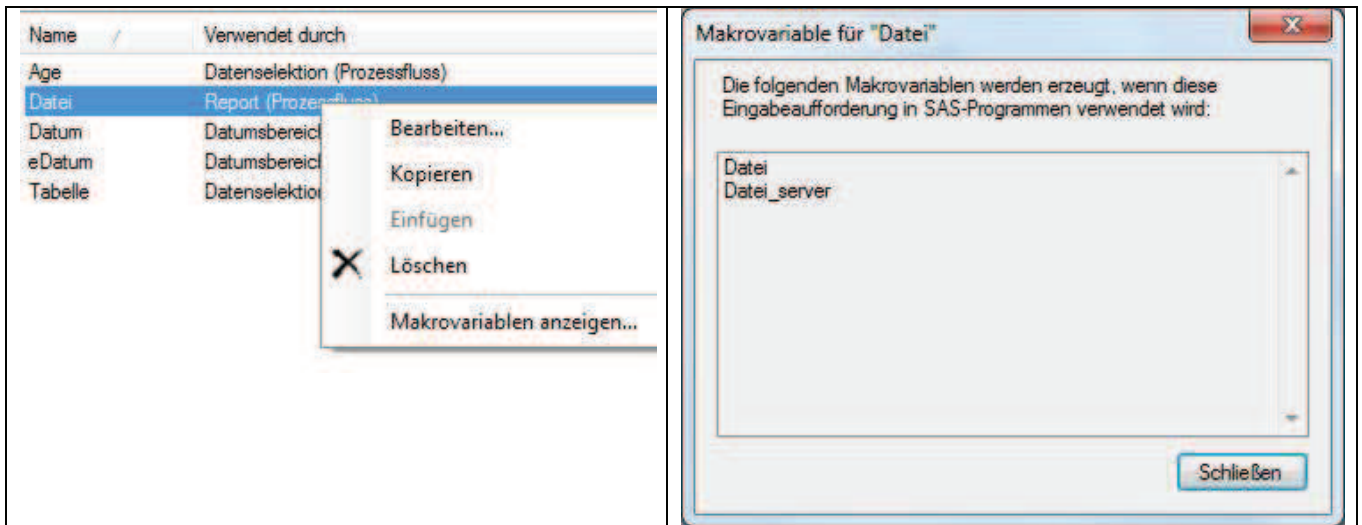


Abbildung 5: Makrovariablenübersicht

Diese Variablen sind standardmäßig nur im Kontext der Programme nutzbar, in denen sie auch zugewiesen wurden. Zum Programmende werden diese mit %syndel wieder entfernt.

### 3.2 Anwendung der Eingabeaufforderungen

Je Programm und Abfrage können die Eingabeaufforderungen hinzugefügt werden. Damit werden dann bei der Ausführung die jeweilig relevanten Parameter abgefragt.

Die Zuweisung zu Programmen erfolgt über die Eigenschaften.

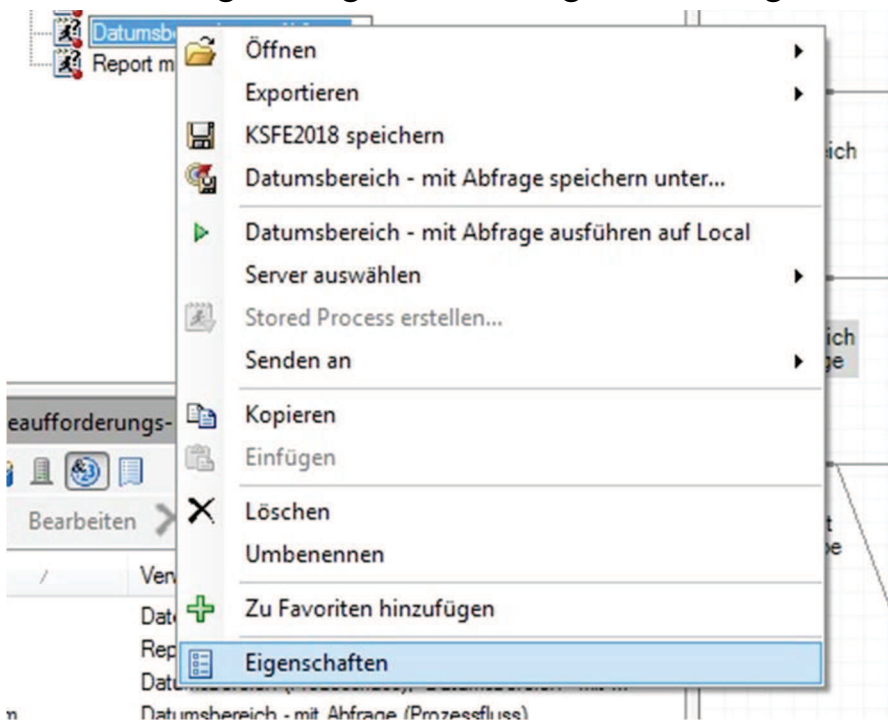


Abbildung 6: Öffnen des Eigenschaften Dialog

Hier können dann die gewünschten (und im Projekt vorhandenen) Eingabeaufforderungen hinzugefügt werden.

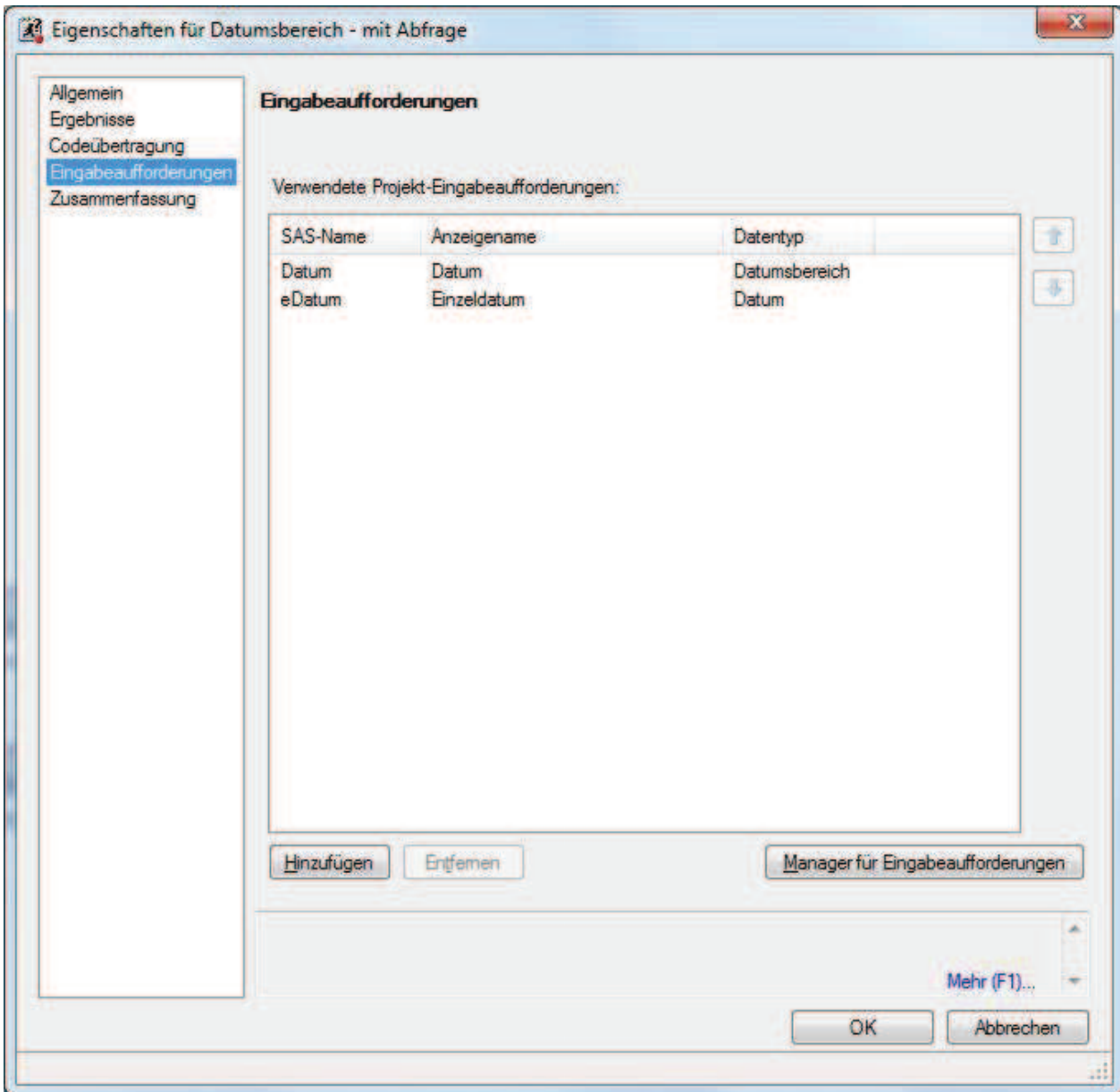


Abbildung 7: Zuweisung einzelner Eingabeaufforderungen

Die Verwendung im Code erfolgt dann analog üblicher Nutzung von Makrovariablen.

```
data work.testdata2;
  format date ddmmyyp10.;

  do date="&datum_min."D to "&datum_max."D;
    if date="&eDatum."D then
      Var="Treffer";
    else Var= " - ";
    output;
  end;
run;
```

## 4 Nutzung für Zugangsdaten

Mit dem bisher erreichten gibt es nun die Möglichkeit, ein Libname oder Filename mit Benutzernamen und Passwort zu versehen. Wird jedoch für das Passwort eine einfache Texteingabe verwendet, ist dieses zum einen bei der Eingabe aber auch in der Makrovariable klartextlich zu sehen.

Wird in den Eigenschaften „Maskierte Einzelzeile“ selektiert, wird mit einem kurzen Hinweis ermöglicht, den Namen der Variable um `_PASSWORD` zu ergänzen.

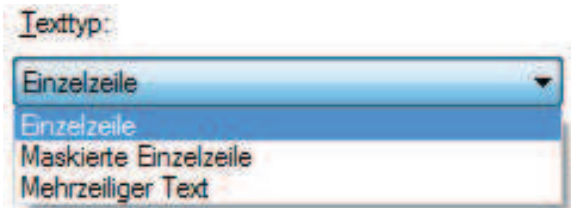


Abbildung 8: Maskierung der Eingabe



Abbildung 9: Ergänzung um `_PASSWORD`

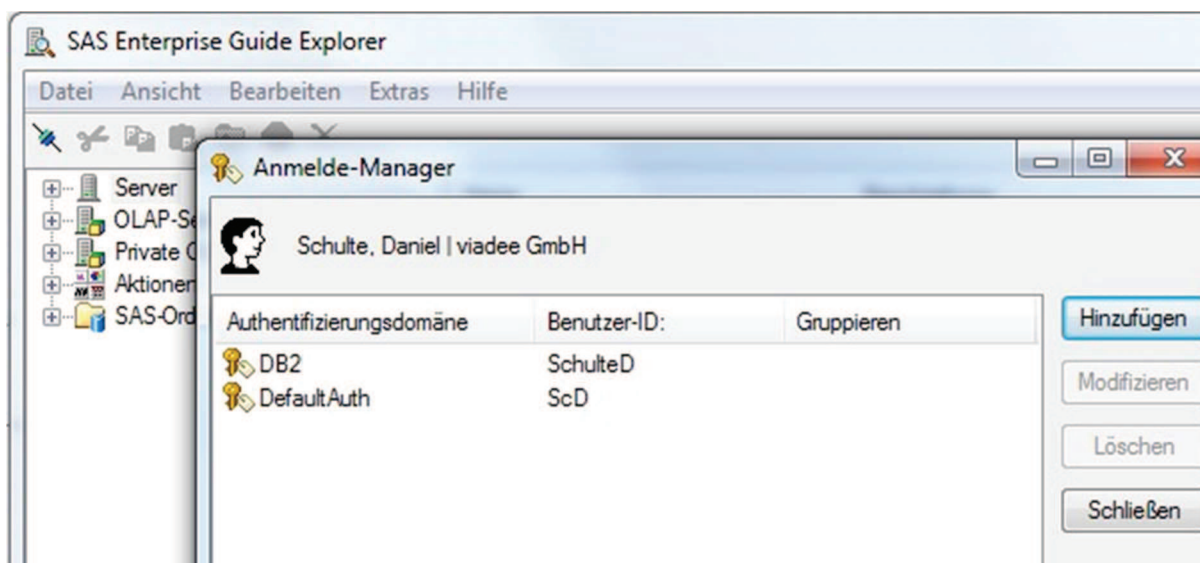
Dies hat dann im Log zur Folge, dass der Inhalt der Texteingabe nun nicht mehr im Log erkennbar wird:

```
10          %LET password_PASSWORD = XXXXX;

30          %put Das Passwort lautet: &password_PASSWORD.;
Das Passwort lautet: {sas002}75F12A4A1812F87907B92E38
```

Aus dieser SAS verschlüsselten Form des Passworts kann man sich nun nicht mehr einfach an anderen Systemen anmelden. Wird diese Zeichenkette allerdings publik, kann diese zumindest aus dem SAS System heraus als Authentifizierung missbraucht werden! Auf einem Standalone System ist dies aber immer noch die beste Lösung. Wenn in einer Client Server Umgebung gearbeitet wird, können über die Metadaten die Zugangsdaten in AuthDomains verwaltet werden.

Über Extras > SAS Enterprise Guide Explorer und dann Datei > „Anmeldenamen verwalten“ lassen sich in den zentralen SAS Metadaten Benutzer und Passwörter hinterlegen.



**Abbildung 10:** Verwaltung von Anmeldenamen

Im Code (z.B. in einem Libname Statement) wird dann über das Schlüsselwort

```
authdomain="DB2"
```

der Bezug hergestellt. Damit wird dann in dem Kontext des aktuell angemeldeten Users in den Metadaten die AuthDomain namens DB2 gesucht und die dort hinterlegten Benutzer und Passwort Informationen werden für die Anmeldung genutzt.

Dies hat den Vorteil, dass in dem Programmcode lediglich die Referenz auf das Metadatenobjekt genutzt wird und in der Ausführung kommen die Rechte zum Tragen, die der jeweilige User besitzt. Wenn also ein sehr weit berechtigter User mit der Datenbank kommuniziert und beispielsweise einen nur für diese Stufe freigegebene Detailtiefe bekommt, wird ein anderer User der die Auswertung ausführt, auch nur den Detailgrad zu sehen bekommen, für den dieser auch berechtigt ist.

Mit den AuthDomains müssen auch nicht bei jeder Abfrage die Passwortinformationen erneut eingegeben werden sondern lediglich einmal in den Metadaten. Die AuthDomains arbeiten dann im Hintergrund.

Eine beliebte Fehlerursache hierbei ist es, beim Passwortwechsel die AuthDomains zu vergessen und dort weiter mit dem alten Passwort zu arbeiten. Durch das nun nicht mehr gültige Passwort entstehen zum einen Fehler bei der Anmeldung und ggf. kann durch mehrfache ungültige Anmeldeversuche ein User gesperrt werden.