

Zoomen in Access, Teil 2

Im ersten Teil unserer Reihe mit dem Titel »Formulare zoomen in Access« (www.access-im-unternehmen.de/1605) haben wir uns den ersten Wurf von Microsoft zum Zoomen in Formularen angesehen. Hier war das Zoomen noch auf wenige Ansichten beschränkt, aber Microsoft hat nun nachgebessert. Auch haben wir bemängelt, dass es aktuell noch keine Möglichkeit gibt, den Zoomfaktor programmatisch abzufragen. Außerdem gab es nur einige wenige Befehle, mit denen wir bestimmte Zoomfaktoren einstellen konnten. Auch hier liefert die neue Version eine Neuerung, denn wir können den Zoomfaktor nun nicht nur per VBA genau setzen, sondern diesen auch abfragen. Diese und weitere Neuerungen, die mit den Versionen 2608 und 2609 veröffentlicht wurden, stellen wir in diesem Beitrag vor.

Voraussetzung

Die hier vorgestellten Funktionen sind erst mit Version 2609 vollständig verfügbar.

- Zoomen per Maus (**Strg + Alt + Mausrad**)
- Einstellen eines anwendungsweiten Standard-Zoomfaktors über die Access-Optionen

Was bisher geschah ...

Im ersten Beitrag zum Thema **Zoomen in Access** haben wir bereits einige neue Funktionen rund um das Zoomen vorgestellt:

- Zoom von Tabellen, Abfragen und Formularen in der Datenblattansicht
- Formulare in der Formularansicht
- Zoomen über die Benutzeroberfläche (50 %, 75 %, 100 %, 125 %, 150 %, 175 %, 200 % und 500 %) und über die Statusleiste
- Zoomen per Tastenkombination (**Strg + Alt + Plus/Minus**)

Nun schauen wir uns an, was sich seit dem letzten Update mit Stand der Version 2609 geändert hat.

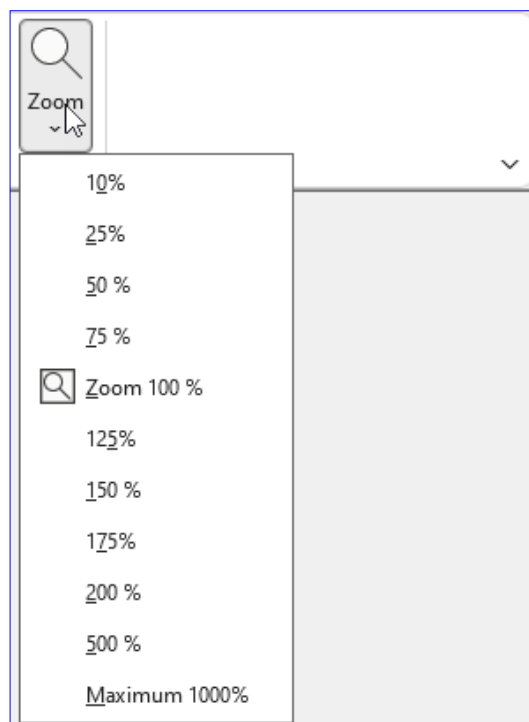


Bild 1: Neue Zoomfaktoren in der Benutzeroberfläche

Nachtrag: Standard-Zoom-Faktor per VBA

Im vorherigen Beitrag haben wir zwar erwähnt, dass wir den Standard-Zoomfaktor mit der Access-Option **Standardzoom** einstellen können. Dies können wir auch per VBA erledigen. Zum Auslesen:

```
Debug.Print Application.GetOption("Default Zoom Level")
```

Zum Einstellen (beispielsweise auf 125 %):

```
Application.SetOption("Default Zoom Level"), 125
```

Neue Zoomfaktoren über die Benutzeroberfläche verfügbar

Die erste Änderung finden wir im Ribbon vor, wenn wir bei geöffnetem Formular die Schaltfläche mit den Zoomfaktoren aufklappen (siehe Bild 1).

Hier entdecken wir neue Zoomfaktoren, nämlich **10 %**, **25 %** und **1000 %**.

Zoomen in der Endlosansicht und in geteilten Formularen

Neben den bisherigen Ansichten werden nun auch die Endlosansicht und geteilte Formulare unterstützt. Das Zoomen gelingt über die gleichen Wege wie bei den übrigen Formularansichten (siehe Bild 2).

Ändern des Zoomfaktors per Mausrad

Bisher haben wir die folgende Kombination zum Ändern des Zoom-Faktors über das Mausrad benötigt (eventuell ist diese Änderung nicht neu – aber wir hatten keine ältere Access-Version mehr zur Verfügung, um dies zu verifizieren):

Strg + Alt + Mausrad

Nun reicht jedenfalls das Betätigen der folgenden Kombination aus:

Strg + Mausrad

Zoomen von Popup-Formularen

Auch für Popup-Formulare steht die Zoom-Funktion nun zur Verfügung. Popup-Formulare sind solche Formulare, die wir zum Beispiel mit dem folgenden

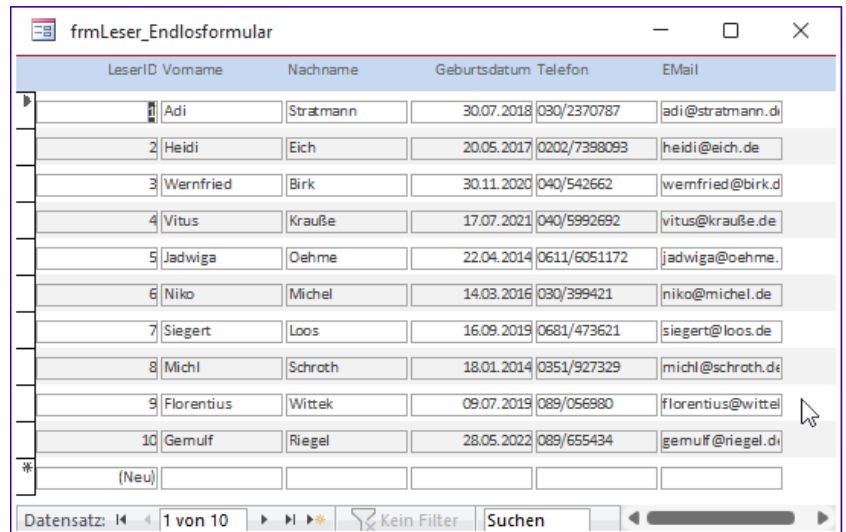


Bild 2: Ein Endlosformular mit einem Zoomfaktor von 50 %

Befehl öffnen – oder für die wir die Eigenschaft **Popup** auf **Ja** einstellen:

```
DoCmd.OpenForm "Formularname", _  
WindowMode:=acDialog
```

Zoomen anwendungsweit aktivieren und deaktivieren

In den Access-Optionen kennen wir aus dem vorherigen Beitrag bereits die Einstellung **Standardzoom**. Hier ist

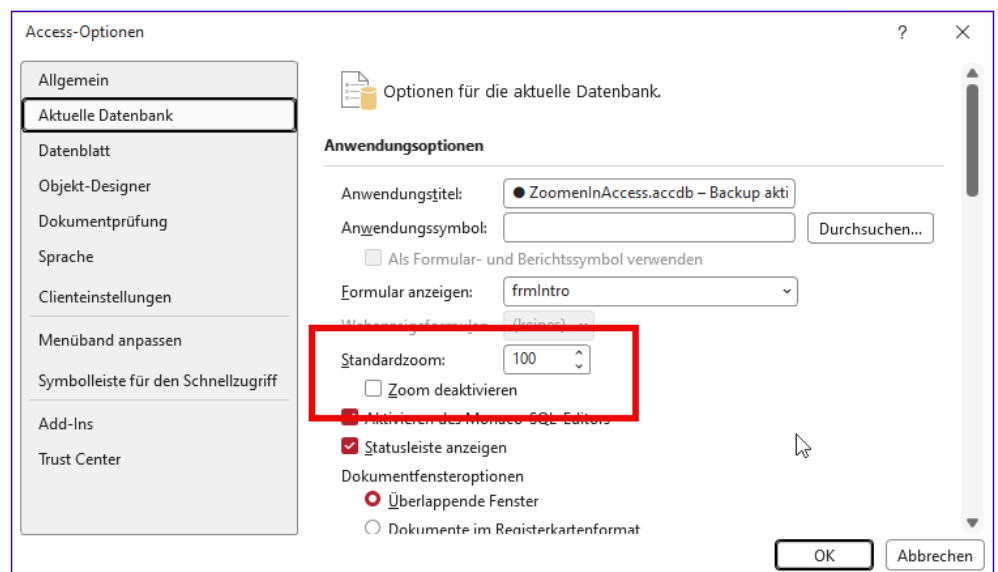


Bild 3: Einstellen des Standard-Zoomfaktors und Aktivieren oder Deaktivieren der Zoom-Funktion

eine neue Option namens **Zoom deaktivieren** hinzugekommen, mit der wir für die aktuelle Anwendung festlegen können, ob der Benutzer den Zoomfaktor einstellen kann oder nicht (siehe Bild 3).

Diese Option können wir auch per VBA auslesen oder setzen. Zum Auslesen verwenden wir die folgende Anweisung:

```
Debug.Print Application.GetOption("Disable Zoom")
```

Um die Option einzustellen, können Sie folgenden Befehl nutzen. Die Übergabe des Werts **True** deaktiviert das Zoomen, **False** aktiviert es wieder:

```
Application.SetOption _
    "Disable Zoom", True
```

Beachten Sie, dass diese Einstellung erst nach dem erneuten Laden der Access-Datenbank wirkt.

Zoomen pro Formular ein- und ausschalten

Wir können die Zoomfunktion aber auch je Formular ein- oder ausschalten. Dazu hat Microsoft eine neue Eigenschaft namens **Zoom zulassen** hinzugefügt, die wir in den Eigenschaften des Formulars unter **Format** finden (siehe Bild 4).

Auch diese Einstellung können wir per VBA vornehmen. Dazu nutzen wir die Eigenschaft **AllowZooming**, die wir allerdings nicht direkt adressieren können, sondern nur über die **Properties**-Auflistung.

Um das Zoomen vom Klassenmodul des Formular aus zu aktivieren oder deaktivieren, können wir die folgenden Anweisungen nutzen – hier zum Deaktivieren:

```
Me.Properties("AllowZooming") = False
```

Zum Aktivieren der Zoom-Funktion nutzen wir diesen Befehl:



Bild 4: Zoom für ein Formular aktivieren oder deaktivieren

```
Me.Properties("AllowZooming") = True
```

Um diese Eigenschaft für ein Formular von außen einzustellen, ersetzen wir **Me** durch die entsprechende Referenz, zum Beispiel:

```
Forms!Formularname.Properties("AllowZooming") = True
```

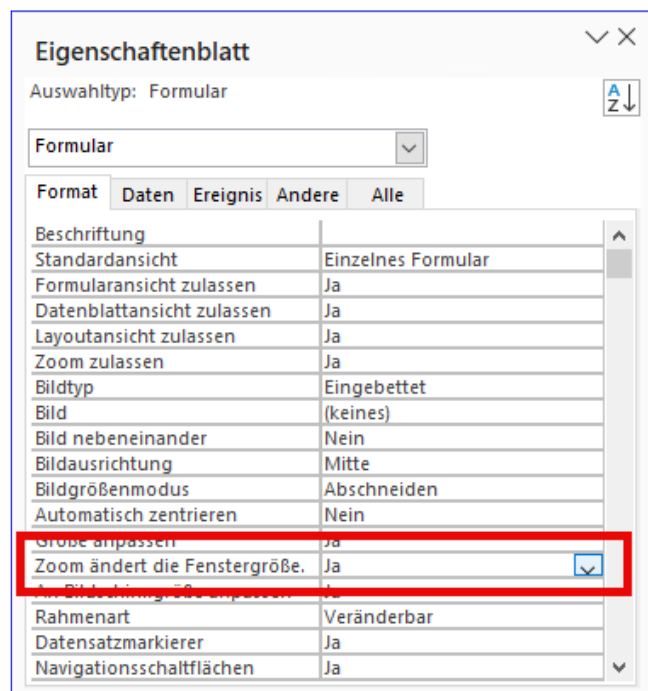


Bild 5: Anpassen der Fenstergröße beim Zoomen aktivieren

Oder, wenn wir die Eigenschaft für das aktuell fokussierte Formular einstellen wollen:

```
Screen.ActiveForm.Properties("AllowZooming") = True
```

Wenn die Eigenschaft auf **False** eingestellt ist, können wir den Zoomfaktor übrigens auch nicht mehr über die Benutzeroberfläche einstellen – die entsprechende Schaltfläche ist dann deaktiviert.

Fenster beim Zoomen vergrößern oder verkleinern

Und es wird noch besser. Bisher war es so, dass der Inhalt eines Formulars zwar gezoomt werden konnte. Wenn wir den Inhalt allerdings vergrößert haben, konnten wir nicht mehr alle Inhalte sehen, da sich das Formular nicht mit vergrößert hat.

Dies hat sich nun geändert. Das Verhalten müssen wir allerdings erst noch aktivieren, was wir für jedes Formular einzelnen durchführen müssen.

Dazu stellen wir eine weitere neue Formulareigenschaft namens **Zoom ändert die Fenstergröße** auf den Wert **Ja** ein (siehe Bild 5).

Wenn wir den Zoomfaktor nun anpassen, ändert sich die Fenstergröße entsprechend dieses Faktors.

Auch diese Eigenschaft können wir per VBA auslesen und einstellen. Das Auslesen gelingt mit dem folgenden Befehl, hier wieder für den Aufruf innerhalb des Formularmoduls:

```
Debug.Print Me.Properties("ZoomResizesWindow")
```

Das Einstellen dieser Eigenschaft können wir wie folgt erledigen – hier zum Aktivieren der automatischen Größenänderung:

```
Me.Properties("ZoomResizesWindow") = True
```

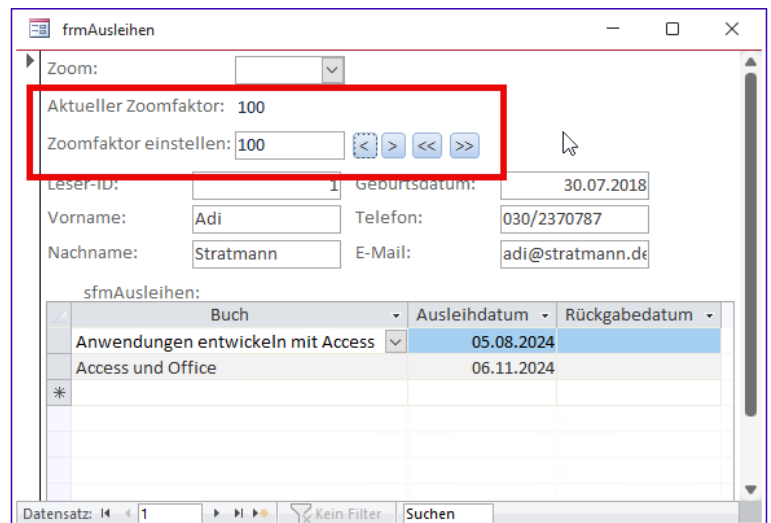


Bild 6: Steuern des Zoomfaktors über die Steuerelemente eines Formulars

Zoomfaktor per VBA einstellen und auslesen

Bisher hatten wir nur verschiedene **RunCommand**-Konstanten, um den Zoomfaktor per VBA einzustellen – hier zum Beispiel zum Einstellen auf 75 %:

```
Application.RunCommand acCmdZoom75
```

Nun können wir über eine neue Eigenschaft namens **Zoom** den Zoomfaktor nicht nur einstellen, sondern auch auslesen.

Auch diese Eigenschaft müssen wir über die **Properties**-Auflistung adressieren.

Um den aktuellen Zoomfaktor eines Formulars auszulesen, verwenden wir den folgenden Befehl:

```
Debug.Print Me.Properties("Zoom")
```

Um den Zoomfaktor einzustellen, nutzen wir zum Beispiel:

```
Me.Properties("Zoom") = 120
```

Wir können hier beliebige Werte zwischen 10 und 1.000 verwenden. Es sind auch größere oder kleinere Werte möglich, ohne dass ein Fehler ausgelöst wird, aber die

Grenzen für den tatsächlichen Zoomfaktor liegen bei 10 % und 1.000 %.

Im Beispielformular **frmAusleihen** haben wir einige neue Textfelder und Schaltflächen hinzugefügt, mit denen wir mit dem Zoomfaktor experimentieren können (siehe Bild 6).

Schauen wir uns zunächst das Textfeld **txtZoomfaktor-Einstellen** an. Für dieses haben wir die folgende Ereignisprozedur hinterlegt, die durch das Ereignis **Nach Aktualisierung** ausgelöst wird:

```
Private Sub txtZoomfaktorEinstellen_AfterUpdate()  
    Me.Properties("Zoom") = Me.txtZoomfaktorEinstellen  
    Me.txtAktuellerZoomfaktor = Me.Properties("Zoom")  
End Sub
```

Die Prozedur legt den eingetragenen Zahlenwert für die Eigenschaft **Zoom** fest und trägt diesen außerdem in das Textfeld **txtAktuellerZoomfaktor** ein.

Damit dieses Textfeld gleich beim Öffnen des Formulars mit dem aktuellen Zoomfaktor gefüllt wird, verwenden wir die folgende Prozedur für das Ereignis **Beim Laden** des Formulars:

```
Private Sub Form_Load()  
    Me.txtAktuellerZoomfaktor = Me.Properties("Zoom")  
End Sub
```

Außerdem haben wir vier Schaltflächen hinterlegt, mit denen wir den Zoomfaktor jeweils um ein Prozent oder um zehn Prozent verringern oder vergrößern können.

Dazu subtrahieren wir entweder den Wert 1 oder 10 vom aktuellen Wert oder addieren diesen hinzu:

```
Private Sub txtZoomMinus1_Click()  
    Me.Properties("Zoom") = Me.Properties("Zoom") - 1  
End Sub
```

```
Private Sub txtZoomMinus10_Click()  
    Me.Properties("Zoom") = Me.Properties("Zoom") - 10  
End Sub
```

```
Private Sub txtZoomPlus1_Click()  
    Me.Properties("Zoom") = Me.Properties("Zoom") + 1  
End Sub
```

```
Private Sub txtZoomPlus10_Click()  
    Me.Properties("Zoom") = Me.Properties("Zoom") + 10  
End Sub
```

Hier fehlt nur noch, dass unser Textfeld **txtAktueller-Zoomfaktor** auf den aktuellen Wert eingestellt wird. Dies könnten wir in jeder der Prozeduren durch eine eigene Anweisung erreichen:

```
Me.txtAktuellerZoomfaktor = Me.Properties("Zoom")
```

Aber wir wollen auch sicherstellen, dass der Zoomfaktor bei Änderungen über die Benutzeroberfläche oder per Tastenkombination angepasst wird. Dazu haben wir die Eigenschaft **Zeitgeberintervall** auf den Wert **100** eingestellt, sodass die folgende Ereignisprozedur alle 100 Millisekunden ausgelöst wird:

```
Private Sub Form_Timer()  
    Me.txtAktuellerZoomfaktor = Me.Properties("Zoom")  
End Sub
```

Standarwerte der Zoom-Eigenschaften in Formularen

Wenn wir ein neues Formular anlegen, hat die Eigenschaft **Zoom zulassen** den Wert **Ja** und **Zoom ändert die Fenstergröße** weist den Wert **Nein** auf.

Das Gleiche gilt für alle bestehenden Formulare. Wer sich hier wünscht, dass die Zoom-Funktion für alle Formulare deaktiviert wird, kann dies für die aktuelle Datenbank einfach über die entsprechende Einstellung in den Access-Optionen einstellen.

Wer möchte, dass alle Formulare beim Zoomen grundsätzlich auch die Formulargröße ändern, muss die Anpassung je Formular selbst vornehmen.

Dazu muss man entweder alle Formular einmal öffnen, die Eigenschaft ändern und die Formulare speichern und schließen, oder man erledigt dies per VBA. Wie das gelingt, zeigen wir im Beitrag **Formulareigenschaften global ändern** (www.access-im-unternehmen.de/1615).

Wenn wir aber nur sicherstellen wollen, dass alle Formulare das gleiche Verhalten bezüglich der Zoomfunktion zeigen und dieses Verhalten je nach Situation anpassen wollen, können wir auch einfach zwei globale Konstanten deklarieren:

```
Public Const cBolAllowZooming As Boolean = False  
Public Const cBolZoomResizesWindow As Boolean = False
```

Die Werte rufen wir dann in der Prozedur ab, die durch das Ereignis **Beim Laden** ausgelöst wird, und weisen diese den entsprechenden Eigenschaften zu:

```
Private Sub Form_Load()  
    Me.txtAktuellerZoomfaktor = Me.Properties("Zoom")  
    Me.Properties("AllowZooming") = cBolAllowZooming  
    Me.Properties("ZoomResizesWindow") = _  
        cBolZoomResizesWindow  
End Sub
```

Auf diese Weise werden alle Formulare mit dem gewünschten Modus geöffnet.

Zoomen von Berichten in der Berichtsansicht
Berichte in der Seitenansicht konnten wir bereits früher zoomen.

Für Berichte in der Berichtsansicht steht nun ebenfalls die Zoom-Funktion zur Verfügung.

Hier konnten wir aber die Eigenschaften **Zoom zulassen** oder **Zoom ändert die Fenstergröße** (noch) nicht finden.

Die VBA-Eigenschaft **Properties("Zoom")** konnten wir allerdings schon erfolgreich einsetzen.

Achtung: Properties lösen in älteren Access-Versionen Fehler aus

Ein wichtiger Hinweis: Die neuen Eigenschaften für Formulare wie **Zoom**, **AllowZooming** und **ZoomResizesWindow** sind in älteren Access-Versionen nicht vorhanden.

Wenn Sie diese nutzen, sollten Sie diese durch eine entsprechende Fehlerbehandlung absichern, da der Aufruf über die **Properties**-Eigenschaft sonst Laufzeitfehler auslöst.

Zusammenfassung und Ausblick

Die aktuelle Erweiterung der Zoom-Funktion für Formulare ist bereits brauchbar.

Was noch nicht funktioniert:

- Anpassung von Steuerelementen wie **TreeView** oder **Listview** (hier wird nur die Größe entsprechend angepasst)
- Der Bereich mit den Navigationsschaltflächen wird noch nicht angepasst.
- Und wünschenswert wäre ebenfalls ein Ereignis, das beim Ändern des Zoom-Faktors ausgelöst wird.