



SAP GUI für Windows und ActiveX

Performance

Stefan Schnell
11. April 2011



Performance mit der Winkelfunktion Sinus

Weitere Performance-Betrachtungen

Sonstiges und Fragen



Performance mit der Winkelfunktion Sinus

Weitere Performance-Betrachtungen

Sonstiges und Fragen



Ursächliche Fragestellung

Kann durch die Verwendung
schneller (Assembler-)Routinen
auf dem Präsentationsserver
eine Laufzeitverbesserung
von ABAP-Programmen erreicht werden?



Vorgehensweise

Zum Test wurde die Sinus-Berechnung eines Winkels in Grad gewählt.

Es wurde eine Assembler-Routine entwickelt und diese der ABAP-Funktion gegenübergestellt.



Assembler-Routine

```
Pi180 = 0.017453292519943295769236907684886
```

```
Function FSin(ByVal AngleDeg As Single) As Single
```

```
    ! FNINIT  
    ! FLD AngleDeg  
    ! FLD Pi180  
    ! FMULP st(1), st(0)  
    ! FSIN  
    ! FSTP Function
```

```
End Function
```



1. Ergebnis

Das Laufzeitverhalten der Assembler-Routine wurde mit `GetTickCount` vermessen.

Sie zeigte erst ab 100.000 Durchläufen ein spürbares Messergebnis.



ActiveX-Modul

Die Assembler-Routine wurde als Methode in einem ActiveX-Modul eingebunden.



ABAP-Programm

```
Do 1000 Times.  
  AngleRad = 45 * Pi180.  
  Result = sin( AngleRad ).  
EndDo.
```

versus

```
Do 1000 Times.  
  Call Method Of oFSinCOM 'FSin' = Result  
  Exporting #1 = 45.  
EndDo.
```



Versuchsanordnung

Applikations- und Präsentationsserver werden in einer VM betrieben.

Der SAP GUI hat die Version 7.10 und das SAP-System die Version SAP EHP 1 for SAP NetWeaver 7.0.



2. Ergebnis

Messungen mit `Get Run Time Field` zeigten signifikante Unterschiede zwischen den unterschiedlichen Ansätzen.

ABAP-Routine:	$324 * 10^{-6}$ Sek.
ActiveX-Routine:	$9.682.618 * 10^{-6}$ Sek.



Weitere Fragestellung

Was ist die Ursache für den deutlichen Unterschied zwischen den Ansätzen?



Performance mit der Winkelfunktion Sinus

Weitere Performance-Betrachtungen

Sonstiges und Fragen



Vorgehensweise

In unterschiedlichen Testreihen wurden verschiedene Ansätze vermessen und ausgewertet:

- Aufruf einer leeren Routine
- Aufruf einer leeren Routine mit Rückgabewert
- Aufruf einer leeren Routine mit Übergabe- und Rückgabewert



3. Ergebnis

Die Verwendung von Übergabe- und Rückgabewerten zeigten nur einen minimalen Unterschied.

In den weiteren Messungen wurde diese Differenzierung nicht weiter betrachtet.



Vermessungsroutine

```
Function PerfTest()
```

```
    OutputDebugString (Time$)
```

```
    OutputDebugString (Str$ (GetTickCount ()))
```

```
    Sleep 5000
```

```
    OutputDebugString (Str$ (GetTickCount ()))
```

```
    OutputDebugString (Time$)
```

```
End Function
```




4. Ergebnis

Pro Aufruf einer ActiveX-Methode aus einem ABAP-Programm wurde, in der beschriebenen Konstellation, ein Zeitbedarf von

$9,6 * 10^{-3}$ Sekunden

ermittelt.



Fazit (1)

Die Verwendung von ActiveX-Methodenaufrufen aus einem ABAP-Programm bedeutet in jedem Fall einen Performance-Verlust gegenüber implementierten Funktionen.

Für die im Beispiel verwendete Sinus-Funktion führt der ActiveX-Methodenaufruf, trotz schneller Assembler-Befehle, zu einem so hohen Performance-Verlust, dass ein Einsatz sich als nicht praktikabel erwies.



Fazit (2)

Die Verwendung von ActiveX-Methodenaufrufen in ABAP-Programmen sollte durchdacht sein.

Vorhandenen ABAP-Befehlen sollte immer der Vorzug gegeben werden.

ActiveX sollte ausschließlich an den Stellen zum Einsatz kommen, an denen ABAP keine Lösung anzubieten hat.



Fazit (3)

Die Verwendung von ActiveX-Methodenaufrufen in Massenverarbeitungen sollte äußerst kritisch und die Performance intensiv betrachtet werden.



Performance mit der Winkelfunktion Sinus

Weitere Performance-Betrachtungen

Sonstiges und Fragen



Gibt es noch

- **Fragen,**
- **Anregungen** oder
- **Wünsche?**



**Danke
für Ihre
Aufmerksamkeit**

