Eventi con Variant

http://www.vbsimple.net/weird/wrd_03.htm

Difficoltà: 3 / 5

Questo articolo tratta un bug che si verifica **esclusivamente** con Visual Basic 5, anche dopo aver installato l'ultimo Service Pack 3 per Visual Studio 97. Lo strano comportamento riguarda il passaggio per riferimento di dati Variant all'evento di un controllo utente e si verifica soltanto nei progetti compilati.

Sebbene i conrolli utente ed i moduli di classe siano concettualmente uguali, il comportamento anomalo si verifica soltanto nei controlli utente, in un momento ben preciso.
Per dimostrare questo bug utilizzeremo un progetto molto semplice contenente un modulo di classe, un controllo utente ed un form che li utilizzerà entrambi. Il controllo utente ed il modulo di classe saranno strutturalmente identici, a dimostrare che la differenza tra i due esiste.
Il codice di entrambi sarà quindi il seguente:

```
 1. Option Explicit
 2.
 3. Public Event Saluta(ByRef Saluto As Variant)
 4.
 5. Public Sub Salutami()
 6.     Dim varSaluto As Variant
 7.     varSaluto = "ciao"
 8.     RaiseEvent Saluta(varSaluto)
 9. End Sub
10.
11. Public Sub Salutami2()
12.     Dim varSaluto As Variant
13.     varSaluto = "ciao"
14.     RaiseEvent Saluta(CStr(varSaluto))
15. End Sub
16.
```

L'evento in questione è dichiarato alla riga 3, prende il nome di Saluta e riceve, per riferimento, un parametro di nome Saluto e di tipo Variant. Sono presenti due metodi di nome **Salutami** e **Salutami2** che sostanzialmente non faranno altro che lanciare l'evento ***Saluta***; la differenza tra loro è data dal fatto che il primo passa all'evento ***Saluta*** la variabile *varSaluto*, mentre il secondo passa la stringa contenuta nella stessa variabile.

Potrebbe sembrar strana come chiamata, una stringa dove invece va passata una variabile, ma si tratta comunque di un'operazione comunissima, utilizzata spesso quando si desidera che la funzione chiamata non vada ad intaccare dati al suo esterno, ma si desidera comunque ricevere gli effetti dati dal suo richiamo.

È stata aggiunta al controllo utente, che prenderà il nome di *SmileControl*, la seguente routine per impedirne il ridimensionamento, ma non ha alcun effetto con il resto del progetto:


```
17. Private Sub UserControl_Resize()
18.     UserControl.Height = 240
19.     UserControl.Width = 240
20. End Sub
```

Il form che utilizzerà i due moduli conterrà quattro semplici pulsanti di nome **cmdClsVar**, **cmdClsStr**, **cmdCtlVar** e **cmdCtlStr**, ed un'istanza del controllo *SmileControl* appena creato.

I quattro pulsanti richiameranno rispettivamente: i metodi Salutami e Salutami2 dell'istanza **clsSaluta** ed i metodi Salutami e Salutami2 del controllo **SmileControl1**. Il codice del form pertanto si riduce a poche miserevoli righe:

```
 1. Option Explicit
 2. Private WithEvents clsSaluta As Classe1
 3.
 4. Private Sub Form_Load()
 5.     Set clsSaluta = New Classe1
 6. End Sub
 7.
 8. Private Sub Form_Unload(Cancel As Integer)
 9.     Set clsSaluta = Nothing
10. End Sub
11.
```

L'istanza clsSaluta è dichiarata con WithEvents (riga 2) in modo da poterne utilizzare gli eventi, ed è istanziata al caricamento del form (riga 5) e deallocata alla sua chiusura (riga 9).

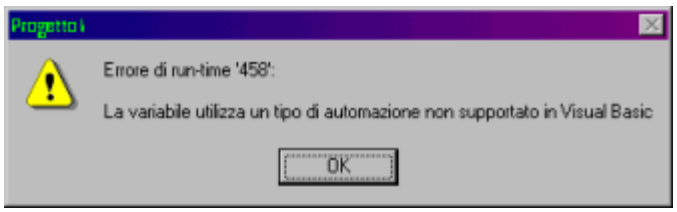
```
12. Private Sub cmdClsVar_Click()
13.     clsSaluta.Salutami
14. End Sub
15.
16. Private Sub cmdClsStr_Click()
17.     clsSaluta.Salutami2
18. End Sub
19.
20. Private Sub cmdCtlVar_Click()
21.     SmileControl1.Salutami
22. End Sub
23.
24. Private Sub cmdCtlStr_Click()
25.     SmileControl1.Salutami2
26. End Sub
27.
```

Come già detto, i quattro pulsanti richiamano i metodi Salutami e Salutami2 del controllo utente e dell'istanza della classe. Lo ricordiamo, il metodo Salutami genera l'evento Saluta passandovi una variabile Variant, mentre Salutami2 genera lo stesso evento ma passandovi invece una stringa.

Concludiamo il nostro progetto con le ultime due routine conclusive:

```
28. Private Sub clsSaluta_Saluta(Saluto As Variant)
29.     MsgBox Saluto
30. End Sub
31.
32. Private Sub SmileControl1_Saluta(Saluto As Variant)
33.     MsgBox Saluto
34. End Sub
```


Anche queste strutturalmente identiche: la prima riguarda l'evento Saluta dell'istanza della classe, mentre la seconda riguarda lo stesso evento ma del controllo utente. Per entrambe è visualizzato un avviso con il valore del parametro **Saluto**, fornito dai metodi Salutami e Salutami2.



Lanciando il progetto all'interno dell'IDE di Visual Basic, tutto funziona regolarmente, ma una volta compilato il progetto e lanciato il file eseguibile, i primi tre pulsanti rispondono regolarmente, mentre il quarto pulsante (***Controllo con String***) genererà l'errore di runtime 458 mostrato nella figura a fianco. L'errore sarà generato in occasione della lettura tramite *MsgBox* dei dati forniti all'evento e non al ricevimento degli stessi alla routine evento.

La descrizione dell'errore è ***"La variabile utilizza un tipo di automazione non supportato in Visual Basic"*** e descrive abbastanza bene l'errore. Infatti, nonostante il metodo Salutami2 avesse passato una stringa all'evento Saluta, quest'ultimo riceve dei dati differenti. È possibile verificarlo sostituendo la riga 33 con:

```
33.     MsgBox VarType(Saluto)
```

La funzione VarType restituirà infatti un valore assai anomalo: 99 decimale, un valore che indica un tipo di dati non previsto nei Variant di Visual Basic. Vedi a tal scopo l'articolo dedicato all'analisi della struttura dei Variant.

Ciò che lascia parecchio perplessi è il fatto che all'interno dell'IDE tutto scorre regolarmente e l'errore è generato soltanto nella lettura di stringhe passate ad eventi di un controllo utente, che si aspettano sia passata una variabile Variant.

L'errore è infatti generato leggendo il valore, il contenuto di ***Saluto*** e non leggendone il suo tipo tramite *VarType* o altro. L'errore non si genera quindi:

- né passando i dati all'evento tramite ByVal
- né al ricevimento di tali dati
- né per i moduli di classe
- né all'interno dell'IDE di VB

Il bug è stato corretto in Visual Basic 6.

Fibia FBI
13 Settembre 2002

Torna all'indice delle Stranezze
