

Computer Telephony Integration Application Bulletin

AB50-0014
June 1, 1998

4170122

Using MS Access 7.0 with StrataLink

This document describes pop-up screen information using Microsoft® Access® 7.0 and StrataLink. Northwind, which comes with Access, is the example database that is used in this bulletin. Northwind demonstrates the methods for writing necessary code for linking StrataLink to Access.

The example code in this bulletin assumes that Access is running with the Northwind database, StrataLink (minimized), and the Customer's Form open.

Operation

StrataLink sends a Dynamic Data Exchange (DDE) command to Access when an outside call is received. Access processes the DDE event as a Structure Query Language (SQL) string, parsing the UPDATE and SET keywords.

Access updates the "Dummy" table first, then makes a call to the SetCallerInfo function, passing the parameter contained in quotes. In the example, StrataLink sends the "&P" token into the "Dummy" table as the Calling Phone number.

Before storing this parameter as a global value, it must be programmed to match the parameter in the table. The number must be stripped of all unwanted characters by calling the StrStrip function. Then the value is reprogrammed to match the information found in the Customer Form.

Popping the form from within the form's timer event prevents a time out from StrataLink. Popping from the SetCallerInfo function takes too much time because the SetCallerInfo searches the database.

However, if your database is fairly small, it might be more efficient to pop the form directly from the SetCallerInfo function. This eliminates keeping the Customer Form open and minimized to check the global DataValid flag twice a second.

When the Customer Form finds the DataValid flag as true, it searches the records for a matching "Phone" field. Once a record is found, it is validated as a match and the form is opened. After opening the form, a final step changes the process to focus on another field to prevent changing the Phone field. You avoid this situation altogether in your application, if you can make the searched field unchangeable.

You can add and change parameter to pass from the phone system to Access using the DDE link. Place the additional parameters in the SetCallerInfo function in the StrataLink action statement.

Toshiba America Information Systems, Inc.

Telecommunication Systems Division
9740 Irvine Blvd., Irvine, CA 92618-1697 (714) 583-3700

1 of 6

Important! *Due to the way Access parses the DDE events, there can be no commas inside the parameter list. A known separator needs to be defined for this function such as '^' character (Shift + 6). Additional parameters could be communicated as follows: ...SetCallerInfo("%P^%N"). This could be used to send both Caller ID Number and Name.*

Access Option Settings

To prevent the “no records have been modified” warning message from occurring in Access for Windows® 95, version 7.0, turn off the confirmation messages for action queries. Viewing the code that is entered (see “DDE Modules Code”), set Access to Full Module View.

► To set Access for full viewing

1. Select Tools, then Options from the Main menu.
2. Click the Edit/Find Tab.
3. In the Confirm grouping, clear the “Action Queries” check box.
4. Click Apply to save the settings.
5. Click the Module tab.
6. In the Code View grouping, place a check in the Full Module View option.
7. Click OK.

Modifications to Your Access Database

Several modifications to the target database are required to get working screen pops in StrataLink. The Northwind database is used as an example; however, a similar approach can be used for any Access database.

Due to the power of the Access language, almost any type of screen pop can be programmed once the basic interface between StrataLink and Access has been created. Follow the steps below to modify your database.

Dummy Table

An SQL statement performs the communication between StrataLink and Access using a dummy table. The DDE interface to Access is somewhat limited because there is not a straightforward way to pass parameters to a subroutine contained in the Access VBA code. Instead we use the SQL statement because it is provided in the DDE interface.

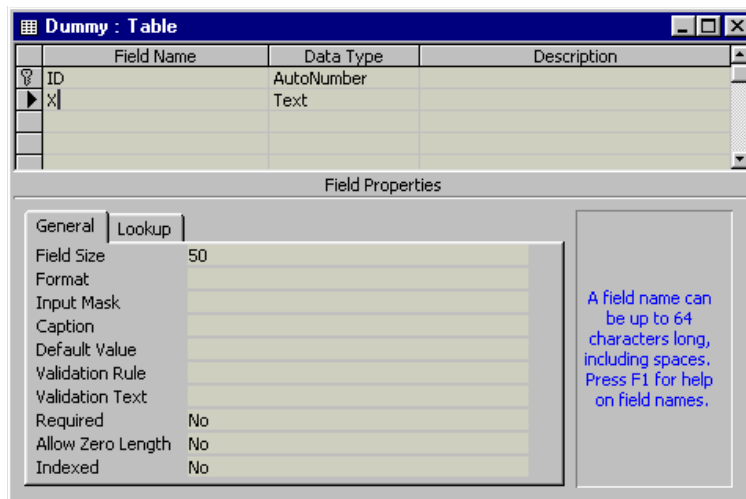
The only way to use the SQL statement is to access a table in some manner. This is where the dummy table comes into action. We use the SQL statement to access the dummy table in a non-intrusive way. As a side effect of the table access, the SQL statement invokes the subroutine we need to get the screen pop started.

► **To set up the table**

1. Click Tables in the Access design-mode window.
2. Click New to create a new table.
3. Click Design View and OK.
4. In the first line of the Field Name column, enter the letter X.
5. Set the data type to “text” (description does not need to be entered).
6. Select the File Menu and Save.
7. When prompted, enter the table name as “Dummy.”

Note You can choose whatever name you want; just remember to make the same name change in all places that this table is used, including the StrataLink Action.

8. Answer Yes to the question about the primary key. This table displays:



9. Close the Dummy Table.
10. Go back to the Tables tab in the Access design-mode window and right click on Dummy Table.
11. Select Field Properties.
12. Set the table as “hidden” so it won’t be seen normally. The Dummy Table disappears.

DDE Modules Code

The DDE Modules code is needed by Access to receive the DDE command from StrataLink and make the parameters available to Access. It is common for most applications that receive the Caller ID number to make this information available. More information can be passed in the DDE link by placing additional parameters in the SetCallerInfo function in the StrataLink action statement.

Access processes the DDE event by parsing the UPDATE and SET keywords. It uses the UPDATE and table name to pass the information and then calls the SetCallerInfo function. Within the SetCallerInfo function are the parameters contained in quotes. In this example, StrataLink sends the Caller ID number using the ‘&P’ token.

The SetCallerInfo function sets the global variables CallerNumber and DataValid. This causes the code, running twice a second in the form, to notice that the DataValid flag needs to be set. The code then attempts to find the phone number based on the information saved in CallerNumber.

A second utility, StrStrip, manipulates the phone numbers so that there is a consistent comparison. To find a match on a field in any database, requires that entries are made to a standard to provide such a match. StrStrip removes all spaces and punctuation from a number. SetCallerInfo uses StrStrip to format the CallerNumber to match that found in the Northwind database.

► **To create the DDE module code**

1. Open the Northwind database design window.
2. Click Modules.
3. Click New to create a new module. The following two lines are already in the form:
Option Compare Database
Option Explicit
4. Enter the code and follow the text in the diagram below.

Note The ‘ character at the beginning of a line indicates comments about the code and is not part of the actual code. For all the other lines, copy the syntax exactly including spacing and capitalization. Access shows different functions in varying colors for ease of reading. Use these to ensure your entry is correct.

```
Option Explicit
Option Compare Database      ‘Use database order for string comparisons

‘Caller information provided by the phone system
Global CallerNumber As String
‘Indicates when the caller information is ready to be used to for database search.
Global DataValid As Integer

Function SetCallerInfo(PhoneParameters As String) As String
    Dim S As String
    ‘Strip and punctuation in the phone number and hyphenate it appropriately.
    S = StrStrip(PhoneParameters, “()-.”)
    If Len(S) = 10 Then
        CallerNumber = “(“ & Left(S, 3) & “) “ & Mid(S, 4, 3) & “-” & Right(S, 4)
    ElseIf Len(S) = 7 Then
        CallerNumber = Left(S, 3) & “-” & Right(S, 4)
    Else
        CallerNumber = S
    End If
    ‘Indicate that we are ready to look up the called information in the database.
    DataValid = True
    ‘Just return a dummy value. This just happens to be the dummy value in the None table
```

```

    '(just in case something goes wrong and the None record is updated).
    SetCallerInfo = "1"
End Function

' *****
' * StrStrip () *
' *****
' Strip any 'Pattern' of characters from a 'String' and return a New string without those
' characters. This is useful for stripping out unwanted characters like "(0-)" out of a tele-
' phone
' number string so that the new string is just digits. For example, with a function call:
'     NewStr = StrStrip ("1(602)555-1234", "(0-)"
' The value in NewStr would be "16025551234".
'

Function StrStrip(OrigStr, PatStr As String) As String
    Dim NStr As String
    Dim Inx, SLen As Integer
    Dim StrC As String * 1           'String (single char)
    NStr = ""                       'Reset New String
    SLen = Len(OrigStr)
    'Loop thru each character in String one at a time
    For Inx = 1 To SLen
        StrC = Mid(OrigStr, Inx, 1)   'Copy Next Char from Original Str
        'Check Pattern looking for Matches
        If InStr(PatStr, StrC) = 0 Then 'Character doesn't match, so Keep It
            NStr = NStr & StrC
        End If
    Next
    StrStrip = NStr
End Function

```

5. Select File and Save.
6. When prompted, name the file "DDEModule."

Forms Code

The code to pop open a form resides in the form itself.

► To add the code to Customer Form

1. Click on Forms in the Access design-mode window.
2. Double click the form you want to pop-up when a call rings (in Northwind, the Customer Form).

3. Enter the following code for the Northwind database:

```

Option Explicit
Option Compare Database
Private Sub Form_Load()
    DataValid = False
    TimerInterval = 500
End Sub

Private Sub Form_Timer()
    Dim SelectedPhone As String
    Dim FoundPhone As String
    If DataValid = True Then
        DataValid = False
        'Look up the phone number in the database.
        DoCmd.GoToControl "Phone"
        DoCmd.FindRecord CallerNumber
        'See if we found the right number. If so, then put up the action to perform request form.
        FoundPhone = Phone.Text
        SelectedPhone = StrStrip(FoundPhone, "()-. ")
        If SelectedPhone = StrStrip(CallerNumber, "()-. ") Then
            DoCmd.OpenForm "Customers"
            DoCmd.OpenForm "Country"
        End If
    End If
End Sub

```

4. Select File and Save All Modules.

StrataLink – Add an Action

The following StrataLink DDE execute string is sent to Access to pop up the form:

```
[RunSQL UPDATE Dummy SET [X]=SetCallerInfo("&P") WHERE X="z";]
```

