

SuperOffice 4.0

Exchanging Data with DDE

Overview

You can transfer data in and out of SuperOffice using Dynamic Data Exchange (DDE). This allows exchanging your data between SuperOffice and any other DDE-capable Windows applications.

Use of the appropriate DDE-calls with SuperOffice is covered and different examples are shown.

Table of Contents

Overview.....	1
Exchanging Data with DDE.....	2
How to use DDE.....	2
Using DDE with SuperOffice.....	3
Data exchange format.....	3
SuperOffice DDE Topics and Service Items.....	4
Descriptions of SuperOffice DDE Topics.....	6
SYSTEM topic.....	6
CURRENT topic.....	7
SQL, SELECT, UPDATE, INSERT, DELETE topic.....	7
CTI - integrating SuperOffice with your phonesystem using DDE.....	9
Setup for outgoing calls using DDE.....	9
Incoming calls.....	9
Extended functions on incoming calls.....	9
SuperOffice DDE Example using Visual Basic.....	10

Exchanging Data with DDE

DDE allows any two Windows applications running on the same machine to communicate between each other. SuperOffice can be one of these applications.

DDE uses a so-called shared memory scheme. In this scheme two applications talk with each other rather like on a telephone call. An application wanting to communicate with another via DDE will dial to the other application. It starts to talk by requesting a so-called service to be performed by the other.

To dial and connect correctly DDE needs the calling application to specify first the name of the application being called, then the name of a topic within that application, and lastly the service wanted.

Normally a DDE-capable application provides several topics. These act like telephone extension-lines with each topic connecting to different services available in the called application.

A special topic called SYSTEM is usually available. SYSTEM supports a special service for listing all the available topics for the application, this is called TOPICS. SYSTEM also supports a special service for listing all the available services for any topic, this is called SYSTEMS.

DDE uses calls of the following types :-

Initiate	to establish a connection
Execute	to ask for a task to be performed
Request	to ask for a value
Poke	to send a value
Terminate	to close a connection (hang up)

How to use DDE

Many Windows applications including Microsoft Word, Excel and SuperOffice support executing DDE calls.

Microsoft Word (using Basic) does this as follows :-

```
channel = DDEInitiate("SUPEROFF", "SYSTEM")
DDERequest(channel, "Topics")
DDETerminate(channel)
```

and Microsoft Excel does it as follows :-

```
=SUPEROFF|SYSTEM!Topics
```

Using DDE with SuperOffice

When communicating with SuperOffice always use the application name SUPEROFF with DDE calls.

Data exchange format

The DDE server in SuperOffice transports data in text format (CF-TEXT). This is a global format, which all DDE applications should support. Regardless of which format the data has inside SuperOffice, it will be converted to text format when passed through DDE links.

SuperOffice DDE Topics and Service Items

Operation	Topic	Item/Command	Notes
Request	System	Topics	Mandatory system item
		Formats	Mandatory system item
		Systems	Mandatory system item
Request	System	Version	Version number
		ArchivePath	Data directory
		NumberOfUsers	Maximum number of users
		SerialNo	Program's serial number
		Company	Program's owner
		CurrentCustomer	Current contact record (Compatible with SO 2.5)
Request	Select	<field>[,<field>,...] FROM <table> WHERE <conditions> Prepares query for retrieving data from any database table.	
		First	Retrieves first record matching the query
		Next	Retrieves next record matching the query
		Free	Releases the query (optional)
		<i>More info on Topic SQL</i>	
Request	Update	<table> SET <field>=<value> [,...] WHERE <conditions> Updates records contents <i>More info on Topic SQL</i>	
		INTO <table> (<field>[,<field>,...]) VALUES (<value>[,<value>,...]) Inserts a new record <i>More info on Topic SQL</i>	
Request	Delete	FROM <table> WHERE <conditions> Deletes records <i>More info on Topic SQL</i>	
		SELECT <field>[,<field>,...] FROM <table> WHERE <conditions> Prepares query for retrieving data from any database table. First Retrieves first record matching the query Next Retrieves next record matching the query Free Releases the query (optional)	
Request	SQL	UPDATE <table> SET <field>=<value> [,...] WHERE <conditions> Updates records contents	
		INSERT INTO <table> (<field>[,<field>,...]) VALUES (<value>[,<value>,...]) Inserts a new record	
		DELETE FROM <table> WHERE <conditions> Deletes records	

Operation Topic		Item/Command	Notes
Execute	System	Maximize	Make SuperOffice use full screen
		Minimize	Make SuperOffice into an icon
		Restore	Make SuperOffice into a Window
		Terminate	Terminates SuperOffice
		Search XXXX	Will search for contact with XXXXXXXXXX as Phonenumbr, give dialog
		SearchX XXXX	Will search for contact with XXXXXXXXXX as phonenumbr, no dialog
		SearchA XXXX	Will search for contact with XXXXXXXXXX as Phonenumbr, no dialog and create new appointment
Request	Current	update	Updates the DDE link with info on Current object in SuperOffice
		<field>	Any template variable stored in the database. Ex. Command "name" will give you the company name of the Current contactcard in SuperOffice

Descriptions of SuperOffice DDE Topics

Each DDE Topic available in SuperOffice is described and their Services are explained.

SYSTEM topic

The SYSTEM topic may be asked for names of its services

```
ch = Initiate("SUPEROFF", "SYSTEM")
systems = Request(ch, "SYSITEMS")
Terminate(ch)
```

This gives the following result:

SysItems, Status, Formats, Help, Version, SerialNumber, Company, ArchivePath, NumberOfUsers, Topics

In addition the System topic supports the following execute commands.

Maximize, Restore, Minimize

All topics used by SuperOffice may be requested using the following DDE command

```
ch = Initiate("SUPEROFF", "SYSTEM")
topics = Request(ch, "TOPICS")
Terminate(ch)
```

This will result in the following list of topics:

Search, Current, Select, Update, Insert, Delete, SQL, System

```
res = Request("SYSTEM", "Status")
res == "Ready" if SuperOffice will accept DDE commands
```

```
res = Request("SYSTEM", "Formats")
res == TEXT SuperOffice only uses TEXT DDE format
```

```
res = Request("SYSTEM", "Help")
Not available
```

```
res = Request("SYSTEM", "Version")
Returns version number
```

```
res = Request("SYSTEM", "SerialNumber")
```

Returns SuperOffice serialnumber. This is used to limit addon packages to only run on one specific installation. Copy protection for 3rd part products.

```
res = Request("SYSTEM", "Company")  
Returns name of owner company of this database.
```

```
res = Request("SYSTEM", "ArchivePath")  
Returns current archivepath. This is useful for creating products that shall also work with SuperOffice Travel, or if your user uses different drive mappings, and you need to know where the SuperOffice archive is.
```

```
res = Request("SYSTEM", "NumberOfUsers")  
Returns the number of users this license contains.  
Also useful for 3rd party products.
```

CURRENT topic

The CURRENT topic is used to read information on current customer. It uses the same abbreviations as merge fields in templates. Before you start reading items, you have to issue an UPDATE. Each time you issue UPDATE values from the current customer are saved for retrieval using DDE.

```
ch = Initiate("SUPEROFF", "CURRENT")  
Request(ch, "UPDATE")  
customerName = Request(ch, "name")  
customerPhone = Request(ch, "phon")  
Terminate(ch)
```

Some values that are available in templates are not available using DDE. These are all values related to the document. These are not available because there is no current document.

Current person and project are only available if they exist. This means that a person is selected for current company, or that a project is selected.

Values from own company and user are always available.

SQL, SELECT, UPDATE, INSERT, DELETE topic

These topics are used for executing SQL statements against a SuperOffice database. You may only use simple statements on one table at a time. Note that when no more data is available SuperOffice returns the text **#EOF#**.

```
ch = Initiate("SUPEROFF", "SQL")
res = Request(ch, "SELECT name FROM Contact")
If res == "OK" Then
  res = Request(ch, "First")
  While res != "#EOF#" Do
    DoSomething(res) -- Process record
    res = Request(ch, "Next")
  End While
  res = Request(ch, "Free")
End If
Terminate( ch )
```

SuperOffice returns **#ACCESS_VIOLATION#** if the user does not have the right privileges.

SELECT is only available for users with userlevel 0, 1, 2 and 3 in SuperOffice.

UPDATE is only available for users with userlevel 0, 1 and 2 in SuperOffice.

INSERT is only available for users with userlevel 0 and 1 in SuperOffice.

DELETE is only available for users with userlevel 0 in SuperOffice.

CTI - integrating SuperOffice with your phonesystem using DDE

SuperOffice has two features to handle phone calls through an external program.

For outgoing calls using the dial button, SuperOffice can issue a DDE execute command to an external DDE server, telling the external program witch number to call.

For incoming calls, SuperOffice can accept a DDE execute command with the calling number, search for the contact that the number belongs to and display the contact card (if the contact is found.)

Setup for outgoing calls using DDE

To make SuperOffice use DDE to make external calls instead of using modem, enter the necessary DDE command in the prefix field in the dial dialog.

Syntax: **DDE;<service>;<topic>;<command>** The phonenumber is appended.

Sample: Using an external DDE server that responds to DDE execute, service=**DIALER**, topic=**SYSTEM**, command=**CALL=xxxxxxx**. (xxxxxx = phone number)

This will give this prefix string in the phone dialog: **DDE;SYSTEM;CALL=**

Incoming calls

An external program can sent a DDE execute command to SuperOffice with the syntax:

Service = **SUPEROFF**
Topic = **System**
Command = **Search xxxxxxxx**
(xxxxxxx = phone number to search for, there must be a space between "Search" and the number)

SuperOffice will then search the database for the number and display a dialog after the following rules:

If the incoming phone number is found in the SuperOffice database SuperOffice will display the name of the contact that is calling and give you a the possibility to go directly to this contact card in SuperOffice.

If the number is not found, then SuperOffice will just display the phonenumber in the dialog.

Phone numbers are searched for in the **search_phone** column in the **phone** table. International prefixes must be removed by the calling application before the command is executed.

Extended functions on incoming calls

Command = **SearchX xxxxxx** Will switch to the contact card, without displaying the dialog.

Command = **SearchA xxxxxx** Will switch to the contact card directly, without displaying the dialog and then activate the dialog for inserting an appointment.

SuperOffice DDE Example using Visual Basic

Example from Visual Basic using CURRENT and SQL topics to insert a new appointment attached to the current customer in SuperOffice.

```
Sub InsertAppointment_Click ()

    LINK_MANUAL = 2
    LINK_OFF = 0

    Dim cuid As String, cname As String, alid As String, associate_id As String
    Dim group_idx As String, SONow As String, overskrift As String

    heading = "Appointment text"

    ' Find date time value. Seconds from 1st January 1970
    SONow = DateDiff("s", "01-01-70", Now)

    ' Get current user and Customer from superoffice
    Text1.LinkTopic = "SUPEROFF|CURRENT"
    On Error GoTo Exit
    Text1.LinkMode = LINK_MANUAL
    Text1.LinkItem = "Update"
    Text1.LinkRequest
    Text1.LinkItem = "cuid"
    Text1.LinkRequest
    cuid = Text1.Text
    Text1.LinkItem = "name"
    Text1.LinkRequest
    cname = Text1.Text
    Text1.LinkItem = "alid"
    Text1.LinkRequest
    alid = Text1.Text
    Text1.LinkMode = LINK_OFF

    Continue = MsgBox("Create appointment for Customer: " & cname, 65, " ?")
    If Continue = 2 Then GoTo Exit

    ' Initiate DDE - SQL communication with SuperOffice
    Text1.LinkTopic = "SUPEROFF|SQL"
    Text1.LinkMode = LINK_MANUAL

    ' Find Text_id for new appointment
    Cmd$ = "SELECT next_id FROM Sequence WHERE id = 31"
    Text1.LinkItem = Cmd$
    Text1.LinkRequest
    Text1.LinkItem = "First"
    Text1.LinkRequest
```

```

textid = Val(Text1.Text)
' Insert the appointment text
Cmd$ = "INSERT INTO Text (type, text) VALUES (4, "" & heading & "")"
Text1.LinkItem = Cmd$
Text1.LinkRequest
' Test if the expected id was used
Cmd$ = "SELECT next_id FROM Sequence WHERE id = 31"
Text1.LinkItem = Cmd$
Text1.LinkRequest
Text1.LinkItem = "First"
Text1.LinkRequest
newId = Val(Text1.Text)
If newId <> textId + 1 Then
    newId = newId - 1
    'Did not get expected id value for Text find the correct one
    While newId > textId Do
        Cmd$ = "SELECT next_id FROM Sequence WHERE id = 31"
        Text1.LinkItem = Cmd$
        Text1.LinkRequest
        Text1.LinkItem = "First"
        Text1.LinkRequest
        If heading = Text1.Text Then
            textId = newId
        End If
        newId = newId - 1
    End While
End If

' Find associate_id for current user
Cmd$ = "SELECT id FROM Associate WHERE loginname = " & alid
Text1.LinkItem = Cmd$
Text1.LinkRequest
Text1.LinkItem = "First"
Text1.LinkRequest
associate_id = Text1.Text
' Find user group for this user
Cmd$ = "SELECT group_idx FROM Associate WHERE id = " & associate_id
Text1.LinkItem = Cmd$
Text1.LinkRequest
Text1.LinkItem = "First"
Text1.LinkRequest
group_idx = Text1.Text
' Insert the appointment
Cmd$ = "INSERT INTO Appointment "
Cmd$ = Cmd$ & "(contact_id, associate_id, group_idx, registered,
    registered_associate_id, done, do_by, task_idx, priority_idx, type, status,
    text_id)"
Cmd$ = Cmd$ & "VALUES (" & cuid & "," & associate_id & "," & group_idx &
    "," & SONow & "," & associate_id & "," & SONow & "," & SONow &

```

```
    ", 8, 2, 2, 3 , " & textid & ")"  
Text1.LinkItem = Cmd$  
Text1.LinkRequest
```

Exit:

```
    Text1.LinkMode = LINK_OFF  
    Exit Sub  
End Sub
```