

English User Manual

Copyright © (1990-2022) SW-Tools ApS Duevej 23 DK-2680 Solrød Strand Denmark Phone: +45) 33 33 05 56 Mail: swtools@swtools.com www: www.swtools.com

Object Linking and Embedding

22/11/01 / 2022-09-01 008.384

Contents

Contents	2
1. Preface	3
1.1. Release notes	4
2. OLE in RAPGEN	5
2.1. Step 1 - Defining a supplier letter	6
2.2. Step 2 - Adding the content of a Microsoft Word document as field	8
2.3. The final supplier letter	11
2.4. Using link instead of embedded	12
3. OLE in IQ/DATAMASTER	13
3.1. Creating a simple query	14
3.2. Adding the OLE object step by step	15
3.3. Using 3 button fields for OLE functionality	16
3.4. How to create the object	17
3.5. How the object can be saved	18
3.6. Performing actions on the object	19
3.7. How to work with the final query	20
4. OLE functions	22
4.1. View of the on-line documentation	23
4.2. General error codes	24
4.3. <u>OleAllocate</u> - Allocate a new object	25
4.4. <u>OleFree</u> - Free the object	26
4.5. <u>OleLinkToFile</u> - Link to an object file of any type	27
4.6. <u>OleEmbedded</u> - Create an embedded object	28
4.7. <u>OleFillObjectMenu</u> - Fill a menu with all registered objects	29
4.8. <u>OleFillVerbMenu</u> - Fill a menu with the object verbs	30
4.9. <u>OleDoVerb</u> - Perform an object verb	31
4.10. <u>OleSave</u> - Save object as embedded into file	32
4.11. <u>OleLoad</u> - Load an embedded object from file	33
4.12. <u>OleGetInfo</u> - Return information about object	34
4.13. <u>OleSetInfo</u> - Set object Information	35
4.14. <u>OleMenuCreate</u> - Create a menu	36
4.15. <u>OleMenuDestroy</u> - Destroy a menu	37
4.16. <u>OleMenuAdd</u> - Add item to a menu	38
4.17. <u>OleMenuSelect</u> - Select from menu at the current cursor location	39
4.18. <u>OleDialogCreate</u> - Standard dialog for creating embedded and linked objects	40
4.19. <u>OleDialogFile</u> - Standard dialog for selecting file name	41
5. Technical specifications	42
5.1. Requirements	43
5.2. Files installed	44
Figure list	45
Index	46

1. Preface

SW-Tools Object Linking and Embedding provide you with a simple way to integrate the content of other Windows applications in the report or query defined in a SW-Tools TRIO application. The manual will refer to Object Linking and Embedding as the short name OLE.

For example, you may want to use the word processing features of Microsoft Word when defining a customer letter, or to be able to play a video sequence when querying the article information etc.

The interface provided in TRIO is simply the ability to define a field, which is marked as OLE. Working with the layout of a report or the form of a query you may choose to link to an existing document or create an embedded document.

A link to an existing Microsoft Word document is only a reference to the file name. If you have a document stored as c:/Microsoft/word/customer.doc you may link directly to it. If the document is changed by another user, not working in the TRIO environment, the object will be up to date the next time the report/query is executed.

If using OLE as an embedded document, the actual content of the document is stored together with the TRIO application. Changes to the document object is therefore made only by TRIO using the object application, and cannot be changed directly from the real application.

Choosing between the linking and embedding depends on how you want to store the content of the objects.

This manual will guide you through samples in both RAPGEN and IQ to show the simple use of OLE objects in TRIO.

1.1. Release notes

The following enhancements has been made to the user interface in SW-Tools RAPGEN/IQ: - New field type OLE in the Free field dialog

Please refer to the samples made in this manual to see the interface changes.

2. OLE in RAPGEN

This chapter will step by step describe how you may simply define a letter which will read and print information from the database including text written in Microsoft Word.

2.1. Step 1 - Defining a supplier letter

We start this sample by defining a simple supplier letter, based on the TRIO demo system.

	New report/lette	er:
Main file		Report no
le Supplier fil	Le 🛨	001 ±
Report name		
Supplier Letter		
1		
User name		
Program type ── ○ <u>R</u> eport	C Print z	zero values
● <u>Letter</u>	<u></u> K	<u>C</u> ancel
Select a main file		

1. Defining the supplier letter

The file 'le' has been selected as main file for the letter and the name changed to 'Supplier Letter'.



2. Adding fields to print on letter

Secondly we insert 3 fields from the main supplier file into the layout.

2.2. Step 2 - Adding the content of a Microsoft Word document as field

To add the content of Microsoft Word to the letter you must first define a field of type OLE 2.0 Object. In this sample we have selected the free field number 10 from the database window.

-	Define free	e field
Field no:	10	
Field name:	Text in Microsof	it Word
Field format:	9,T2	
Type Object		Input field
#0 OLE 2.0) Object 🛃 🛨	None
		<u>OK</u> <u>C</u> ancel

3. Defining the OLE 2.0 Object field

Inserted into the layout and sized accordingly we can select the type of OLE object by clicking with the right mouse button on the field and select the function 'Text...'.

۲	E C	Σ	PR-	A [-		0	TT	95 I	#	to	9			廖 🕏	18	P.										
												La	yout													
	X-per X-per	750	1000	2	54	7							(36,	750,2	190,1	844)	#10	101	Fext in	Mic	rosoft	Word	1			
0	17	1ª	12	14	e.	10	17	18	12	110	111	10	110	134	18	.16	TT.	1.10	179	100	131	133	-33	104	128	28
			4		÷	der.			4		÷	÷	12	÷			ş					deres				ł÷
		1	1	1	1	1		1	1	1	177	1	1	1	1	Ť.	1	1	1	1	1	1	1	1	1	1
				1	Į	1	5	1	3	1	Į	3	1	Jan .	1		S		1	Į	Ţ		1	ļ		Į
								1	+	÷.		÷			Co	penha	agen,	99.	.99.9			· · · · ·				
			1	1	1	1	1	1	10		111	100	111		1	1	1	Ì.	1	1	1	1	1	1	1	1
			l	3	į	5					1	See.	£.,	Sec.	1	1	1	Q				÷	1		Ser.	Į.,
ł			1	1	-	1	-	+	+	1				+		÷	÷	÷	++++	{			÷	÷		÷
1		X	XXXXX	XXXXX	*XXX	XXXXX	****	600	1		1	1	1	1	1		1	1		1	-	1	1	1	1	1
		1 00	XXXX	XXXXX	XXX	XXXXX	кхххэ	1			1	1	1	1		1	1		1		1		1			1
ł		80	CKXXX	XXXXX	XXX	*****	XXXXX				4			-			3+++					·				÷
	100		+	+	+		+	1	+	100	1	100	1777	177	1	ţ	1	1	1777	1777	100	100		1	1	to
			1	1	1	1		1	1		1		1	1	1	1			1	1.1	1			1	1	11
1								Pe	n		- 0					Ş							Lana.		4	ş
t	1-1-			1	1	+	1	C	lor		3	1	1	+	1	1	· · · ·	1				1	1	1	1	1
				1		1		5	lect f	ant			1	1		1					1		1		1	1
	a series	coleras	100		-	1.1.1.1	1000	Ju	stity	and the		1000	1	++++	-	irre		1000				i	din di			ş.,
1.8								B	ine te	front	21										2					
-								B	ing to	back		Dat	abase													
5	upplie	r file						B	ackgro	ound i	tem															
Su	pplier	no			10	#OTex	t in M	SI	adow	۱. 	~	Free	_			2	28 @	Free	-			3	37 @	Free		
Na	me				11	@Free	en e	30) dew	n effe Maart	Cŧ	Free				-	29 @	Free				1	38 @	Free		
ΑØ Γα	ATT				13	@Free	2	B	ack b	order		Free					5U @	Free				1	10 @	Free		
Cu	rrency	code			14	@Free	9	FI	p 180	l degr	ees	Free					32 @	Free					11 @	Free		
Ba	lance				15	@Free		Be	x		1	Free					33 @	Free				1	12 @	Free		
	ree				17	@Free @Free				3	26 0	oFree					35 @	Free				1	14 @	Free		
ä	100				18	@Eree	8				97 6	B.Free					16 0	Free					15 (2)	Free		

4. How to select the requested OLE object type

The first time you select the 'Text...' function for an OLE object a dialog will appear with all installed OLE 2.0 objects. From this list we select the name

	OLE 2.0 Object
 Link to e	xisting file
A&L Expres Media Clip Microsoft E Microsoft G Microsoft W Microsoft W MS PowerP MS PowerP Netscape H	s Graphic xcel 5.0-ark xcel 5.0-diagram raph 5.0 Diagram /ord 6.0-billede /ord 6.0-dokument oint 4.0-dias oint 4.0-præsentation ypertext Document
Name	Word.Document.6

5. Selecting Microsoft Word as object type

Secondly RAPGEN will activate the object server, in this case Microsoft Word, hereby allowing you to use the entire functionality of this application to enter the text in a word processing system.



Dear Supplier,

Your current balance is as of todays date 2/28/98

Balance :

6. Entering the content of the object in Microsoft Word

When the text has been entered the application can be closed and the object content will appear in the layout of the letter.

	0 l ₁	1 ⁸	13	14	15	16	17	_1 ⁸	19	10	11	13	13	114	15	1 ¹⁶	117	18	119	120	121
1-							·}	. <u> </u>	-}	- 			- {						- j		
°			-1		- <u>-</u>	4	- <u>}</u>	1		- <u> </u>					-1			4	- <u>-</u>	1	
<u> </u>						+	÷	+		+		-+			-+			+	- 	+	-
5							1	1						-	C C	openh	lagen,	99	99.9	9 9	-
6						1		1							1				1		
7		1	1			1		1	1	1			1	1	1			1	1	1	
8						1		ļ			1				- <u> </u>			1		1	-
9	<u></u>		3	1	1	12-	3	3	-	<u> </u>	- <u> </u>	_ <u>}</u>	- (4		4			22
11			xxxxx	xxxxx	*****	XXXXX	xxxx	d					- {	+			-}			<u>}</u>	
12	1	T is	xxxxx	xxxxx	xxxxx	xxxx	xxxx	(1		1	1	1		1	1	1	1	1	1	
13		- S	xxxxx	xxxxx	xxxxx	XXXXX	(XXX)	C	1			1	1	1	1		1	1	1	1	82
14			1			1	1	1	1		1			1	1			1	1	1	
15							1												1	1	
16	Dear Su,	pplier	,				+	+													
17 18	Your cu	ment l	alance	eisaso	ftoda	ys date	2/28/	19.8										+			-
19	1.1.1		1		1	1	1	1	1	1			1	1	1		1	1	1	11	
20		1	4		1	1		1	1	1	1		1	1	4			1	1	1	
21	Janipan				I	Salanc	e: १	-99	,999	,999	.99			-				4	+	hand	

7. Layout of letter including the OLE object content

We finally add the balance field on top of the OLE object field. The letter is complete.

2.3. The final supplier letter

When the letter is printed you will get the following:

	Copenhagen,	28.02.98
SCHIERMACHER LTD. BOULEVARD ROYAL 63 LUXEMBOURG		
Dear Supplier,		
Your current balance is as of todays date 2/28/98		
Balance : 20,000.00		

8. The printout of the supplier letter

2.4. Using link instead of embedded

The same sample could be done by linking to an existing document file. We have saved the exact same content into a file named

c:/swtools/supplier.doc

and will now link to it instead of having the object embedded in the report.

Insert the field defined as OLE object into the layout and select the 'Text...' function by clicking with the right mouse button on the field.

In the dialog check mark the 'Link to existing file' and enter the file name.

0	LE 2.0 Object
Link to existing file	
c:\swtools\supplier.doc	
<u> </u>	<u>}</u> rowse
	<u>Q</u> K <u>C</u> ancel

9. Linking to an existing file

The final printout will have the same effect as when embedding the object in the layout. Please note, you cannot switch from embedded to link directly. You first have to remove the field from the layout and then insert it again. Then you may link to an existing file instead of embedding an object.

3. OLE in IQ/DATAMASTER

This chapter will describe how you may add OLE functionality to any type of IQ/DATAMASTER programs. This section only illustrates the use in IQ because its the same when working with DATAMASTER.

We have chosen a sample which will actually act as a supplier query, where the user may actually add any type of OLE object as extra information on the supplier. This gives you an idea of what may be generated as applications in TRIO when using OLE objects.

3.1. Creating a simple query

This sample is based on a simple supplier query **le#1-6**

which has been saved as program 1 in IQ.

🎒 001 Supplier (Query with OLE objects	- 🗆 ×
Supplier no		
Supplier no		
Name		11
Address		
Town	19	1
Currency code		
Balance	17-	

10. Simple supplier query in IQ

3.2. Adding the OLE object step by step

Define a **free field #10** in the form function with the name **'OLE Object on supplier'** marked as **'#O OLE 2.0 Object'**.

-	Define free field
Field no:	10
Field name:	OLE Object on Supplier
Field format:	9,T2
ГТуре ———	
Object	#O OLE 2.0 Object 🛃
	<u> </u>

11. Free field defined as OLE object

Insert the free field #10 at the desired position in the form and size the box accordingly. In this sample we have adjusted the box to fit the query window.

After this you add two calculation lines for reading the OLE object **After read of supplierfile** #14="c:/swtools/",#1,".swo"

OleLoad(#10,#14)

3.3. Using 3 button fields for OLE functionality

Add 3 free fields defined as **'&Create'**, **'&Save'** and **'&Action'**. All the fields has the format **8**. When inserted in the layout, mark the fields as object type **'Button'** clicking the right mouse button on each of the 3 fields.

	boor	- 16 - N2 - 12	N 14 1
Namb			
Address	200000000000000000000000000000000000000		
Iownj)0000000000000000000000000000000000000		
Currency coc	l¢99		
Balance	-99,999,999.99		
7			
	1		
		4	
		George	
		1	
		Long Long Long Long Long Long Long Long	
	Г		
		Pen	•
	<u>~</u>	Pen Color Text	X
		Pen Color Text Select foot	•
	** ***	Pen Color Text Select font	•
		Pen Color Text Select font Justify	plier Query v
Supplier file		Pen Color Text Select font Justify Bring to front Bring to back	plier Query v
Supplier file	xx xxxxxxx	Pen Color Text Select font Justify Bring to front Bring to back Background item	plier Query v Free
Supplier file Supplier no Name	xx xxxxxxx 10 #OOLE Obje 11 &Create	Pen Color Text Select font Justify Bring to front Bring to back Background item Shadow	plier Query v Free Free
Supplier file Supplier no Name Address	10 #OOLE Obje	Pen Color Text Select font Justify Bring to front Bring to back Background item Shadow 3D down effect	plier Query v Free Free Free Free
Supplier file Supplier no Name Address Town	10 #OOLE Obje 10 #OOLE Obje 11 &Create 12 &Save 13 &Action 14 @Free	Pen Color Text Select font Justify Bring to front Bring to back Background item Shadow 3D down effect 3D up effect	plier Query v Free
Supplier file Supplier no Name Address Town Currency code Balance	10 #OOLE Obje	Pen Color Text Select font Justify Bring to front Bring to back Background item Shadow 3D down effect 3D up effect Black border	plier Query v plier Query v Free Free Free Free Free Free Free Button
Supplier file Supplier no Name Address Town Currency code Balance Sum	10 #OOLE Obje 10 #OOLE Obje 11 &Create 12 &Save 13 &Action 14 @Free 15 @Free 16 @Free	Pen Color Text Select font Justify Bring to front Bring to back Background item Shadow 3D down effect 3D up effect Black border	Plier Query V Plier Query V Free Free Free Free Free Eutton Combobox
Supplier file Supplier no Name Address Town Currency code Balance Sum @Free	XX XXXXXXX 10 #OOLE Object 11 &Create 12 &Save 13 &Action 14 @Free 15 @Free 16 @Free 17 @Free	Pen Color Text Select font Justify Bring to front Bring to back Background item Shadow 3D down effect 3D up effect Black border Box	Free Free Free Free Free Free Eutton <u>C</u> ombobox Editbox
Supplier file Supplier no Name Address Town Currency code Balance Sum @Free @Free	10 #OOLE Obje 1 &Create 12 &Save 13 &Action 14 @Free 15 @Free 16 @Free 17 @Free 18 @Free	Pen Color Text Select font Justify Bring to front Bring to back Background item Shadow 3D down effect 3D up effect Black border Box Field settings	plier Query v plier Query v Free Free Free Free Free Editbox Listbox
Supplier file Supplier no Name Address Town Currency code Balance Sum @Free @Free	10 #OOLE Obje 11 &Create 12 &Save 13 &Action 14 @Free 15 @Free 16 @Free 17 @Free 18 @Free	Pen Color Text Select font Justify Bring to front Bring to back Background item Shadow 3D down effect 3D up effect Black border Box Field settings	plier Query v plier Query v Free Free Free Free Button Combobox Editbox Listbox

12. Buttons to control the OLE object

3.4. How to create the object

The calculations needed to create an object as embedded or linked may simply be by the following calculations:

Calculations:001 Supplier Query with OLE objects			
By click on a field	± #11 &Create	KWW005 重	
<pre>1 #15-0leDialogCreate(#10,"ENG") 1 if #15-1 BleLinkToFile(#10,"") 1 if #15>1 BleEmbedded(#10,1,"",#15-1,0) 1 1</pre>	 OleCreateValue=OleDialogCreate(#OOLE Object if OleCreateValue=1 OleLinkToFile(#OOLE Object if OleCreateValue>1 OleEnbedded(#OOLE Object 	on supplier,"EHG") ect on supplier,"") t on supplier,1,"",0	

13. Calculations for creating an OLE object

The free field #15 used to retrieve the selected object is defined as format '9,'.

3.5. How the object can be saved

To enable the user to save the created object, we have added the following calculations:

9	Calculations:001 Supplier Query with OLE objects	R A
By click on a field	± #12 &Save	KWW006 ±
1_ #14 = "c:/swtools/",#1,".swo" 0leSave(#10,#14) 1_ 1_ 5_	+ OLEFILeName - "c:/swtools/",Supplier no.".swo" OleSave(#00LE Object on supplier,OLEFileName)	

14. Calculations for saving the created OLE object

The default path **c:/swtools/** and the **supplier no** plus the extension **.swo** generated the complete object file name. For example, for supplier no 205 the file name will be

c:/swtools/205.swo

The format of the free field 14 is **128** (must be an alpha numeric field to hold a file name including the path).

3.6. Performing actions on the object

Because an OLE object may be of different types, the action a user may perform on the object can only be determined by the object application. For example, it an object is of type Microsoft Word, a user may normally only edit the object, where as for an Microsoft Video object, the user may not only edit the content of the object but also play it. The calculations added are a standard sample of how to let the user select the action to perform on an object of any type.

Calculations:001	Supplier Query with OLE objects	
By click on a field	13 &Action KW	W007 ±
<pre>#16-OleMenuGreate() # OleFillVerbMenu(#10,#16,2000) # #17-OleMenuSelect(#16) /* Let user select verb OleMenuDestroy(#16) /* Destroy the menu #18-GETINF0(1) /* Get IQ Application Window #19-GETINF0(6,"#10") /* Get object start x position #20-GETINF0(7,"#10") /* Get object start y position #21-GETINF0(8,"#10") /* Get object end x position #22-GETINF0(9,"#10") /* Get object end y position 0leDoVerb(#10,0,#17,2000,#18,#19,#20,#21,#22) #</pre>	DLEMenu-OleMenuCreate() OLEFillVerbMenu(#00LE Object on supplier,OLEMenu,2000) OLEMenuIten-OleMenuSelect(OLEMenu) /* Let user select werb OleMenuDestroy(OLEMenu) /* Destroy the menu Application Window=GETINF0(1) /* Get 10 Application Window Start X=GETINF0(6,"#10") /* Get object start x position Start Y=GETINF0(6,"#10") /* Get object start y position End X=GETINF0(8,"#10") /* Get object end x position End Y=GETINF0(9,"#10") /* Get object end y position OleDoVerb(#00LE Object on supplier,0,OLEMenuIten,2000,Application	ation Wi

15. Calculations for performing actions on the OLE object

You may refer to the description of the functions used. The free fields used are defined as the following:

Field no	Name	Format
16	OLEMenu	9,
17	OLEMenuItem	9,
18	Application Window	9,
19	Start X	9,
20	Start Y	9,
21	End X	9,
22	End Y	9,

3.7. How to work with the final query

When working with the defined query, it is with the normal functionality of IQ, but extended with the 3 special calculated buttons.

Let say the user locates the supplier no 205, by entering the value 205 in the query key field. The user now wants to write a note on this supplier using Microsoft Word. The first thing the user may do, is to click on the button **Create** which will present the dialog for creating an object. From this dialog the user selects the 'Microsoft Word' as an embedded object.

= 001	Supplier Query with OLE objects	-
Supplier no	205	
Supplier no	205	
Name	SCHIERMACHER LTD.	
Addr	OLE 2.0 Object	
	to existing file	
Curr Microso	ft Word 6.0-dokument	+
MS Pow Netscaj Packag Paint SI Paintbri QuickTi QuickTi Sound	verPoint 4.0-præsentation pe Hypertext Document nop Pro Image ush Picture me Movie me Picture	
Name	Word.Document.6	
	<u>0</u> K	<u>C</u> ancel
	te <u>S</u> ave <u>A</u> ction	ן נ

16. User creating an object for supplier no 205

When the object has been created the user may activate the object application Microsoft Word by clicking on the button **'Action'** and select the **Edit** function.

Times New Roman 👤 18	···
A . I . I . I . 2 . I . 3	• 1 • 4 • 1 • 5 • 1 • 6 • 1 • 7 • 1 • 8 • 1 • 9 • 1 • 10 • 1 • 11 • 1 • 12 • 1 • 13 • 1 • 14 • 1 •
The supplier	• 205 note written in Microsoft Word 6.0
	The man and the first of the co

17. User entering the note in Microsoft Word

When the user closes Microsoft Word the content of the object is displayed within the TRIO IQ query.

- 001 :	Supplier Query with OLE objects	•
Supplier no	205	
Supplier no	205	
Name	SCHIERMACHER LTD.	
Address	BOULEVARD ROYAL 63	
Town	LUXEMBOURG	
Currency coc	le 1	
Balance	20,000.00	
The supplier 2<i>05 note</i> written in Microsoft Word 6.0		
<u>C</u> reat	e <u>S</u> ave <u>A</u> ction	

18. IQ Query with user written note in Microsoft Word

4. OLE functions

The collection of OLE functions are installed as a Windows DLL (Dynamic Linked Library). All the functions are described and implemented as sub functions in TRIO in the 10.xxx file, where xxx is the language code, located in the TRIO installation directory.

4.1. View of the on-line documentation

The on-line description of the OLE function may be viewed directly from the database window. Select the file named 'OLE functions', which is file id 10.

Database		_DX
10 OLE 2.0 Object)	
1 OleAllocate - Allocate a 2 OleFree - Free the objec 3 OleLinkToFile - Link to 4 OleEmbedded - Create an 5 OleFillObjectMenu - Fill 6 OleFillObjectMenu - Fill a 7 OleDoVerb - Perform an o 8 OleSave - Save object as 9 OleLoad - Load an embedd 10 OleGetInfo - Return info	11 OleSetInfo - Set object 12 OleMenuCreate - Create a 13 OleMenuDestroy - Destroy 14 OleMenuAdd - Add item to 15 OleMenuSelect - Select 1 16 OleDialogCreate - Standa 17 OleDialogFile - Standard	

19. On-line documentation of the functions

4.2. General error codes

The following list contain all general error codes:

00 No error 01 Missing OLEx.DLL 02 Missing function in DLL 03 Cannot set message queue 04 Subcall of function failed 05 Call of OLE function failed 06 Illegal OLE version 07 Must call ole_init first 08 Windows register function failed 09 Missing COMPOBJ.DLL 10 Missing STORAGE.DLL 11 Object does NOT support Unknown interface 12 Missing interface

13 Cannot allocate needed memory

14 No verbs found for this object

15 Unknown verb or menu id

16 No objects for OLE2.0

17 Unknown object or menu id

18 No storage allocated for object

19 No call has been made to ole_init

20 Missing OLE2DISP.DLL/OLEAUT32.DLL

21 Cannot convert string to OLE string or visa versa

4.3. <u>OleAllocate</u> - Allocate a new object

This function allocates a new OLE object. The return value is used as first parameter for almost all other OLE functions.

When the object is no longer needed it should free the memory used by calling the function OleFree().

Please note, a OLE field inserted in the layout of a report or in a form on a query/data entry program is automatically allocated by start and will free memory by termination.

OleFree(#50) /* Free memory used by object

4.4. <u>OleFree</u> - Free the object

This function will free the memory allocated by the OleAllocate() function. Please note, a OLE field inserted in the layout of a report or in a form on a query/data entry program is automatically allocated by start and will free memory by termination.

See OleAllocate

4.5. <u>OleLinkToFile</u> - Link to an object file of any type

This function can make a link to an existing file of any type and hereby load the object. For example, if the filename given in *par2* is "notice.doc" the function will let OLE search the registry for the extension ".doc" and find that this is a document of type 'Microsoft Word'. The function will clear the current content of the object *par1* before making the link.

With the function OleDialogCreate() you may simplify the creation of linked and embedded objects my means of a standard Windows dialog.

When using a linked object in TRIO, only the filename will be saved when used in layout/form. This also applies for any files saved with OleSave(). If it is required to have a separate object, owned by the actual TRIO application, use an embedded object instead.

Note 1 If *par2* equals "" (no filename) the function will use the current selected file name. For example, if a prior call to OleDialogCreate() has been made and the user has selected the file named "sheet.xls", it is the current selected file name.

See OleAllocate

4.6. <u>OleEmbedded</u> - Create an embedded object

This function will create an embedded object, e.g. an object owned by the TRIO application. Before an embedded object can be created it is required to pass the object program id. The program id is known from the Windows registry (Refer to the object documentation). For example, to create a Microsoft Word 6.0 document, the program id is "Word.Document.6". The parameters *par4* and *par5* is only used when a prior call has been made to OleDialogCreate() or OleFillObjectMenu()/OleMenuSelect(). Please refer to the documentation on these functions.

OleEmbedded(#50,0,"Word.Document.6",0,0)

4.7. <u>OleFillObjectMenu</u> - Fill a menu with all registered objects

This function may be used when creating a tracking menu for the user to select an object. The function will add items to a menu created by OleMenuCreate().

Each item added to the menu will have a unique id, from 0 to x. Because this may conflict with other menu items added prior to this function call, you may use *par2* as a menu offset, e.g. if *par2* is 1000 the id's of the added items are numbered 1000 to x.

The number of items added depends on how many applications has been installed on the system, which may a lot of objects. Due to display limitations of a tracking menu, a maximum number of items may be given as parameter *par3*, e.g. if *par3* equals 10 a sub menu is created for each 10 items. The sub menu will be names xxxxx 1-10, xxxxx 11-20 etc., where the xxxxx must be given in *par4*.

OleFree(#50) /* Free memory used by object

4.8. <u>OleFillVerbMenu</u> - Fill a menu with the object verbs

This function may be used to fill a menu with all verbs of an object. A verb is the action that may be performed on the object. For example, a verb may be Edit, Open, Play, etc., dependent on the object type. Normally, all objects have the verbs Edit and Open, where some objects have additional verbs like Play (Sound/Video).

The verb is uniquely identified by a number, only known to the object. By using this function it is possible to add items to a menu, where from the user may select the action to perform on the object.

Each item added to the menu will have a unique id, from 0 to x. Because this may conflict with other menu items added prior to this function call, you may use *par3* as a menu offset, e.g. if *par3* is 1000 the id's of the added items are numbered 1000 to x.

OleDoVerb(#50,0,#52,2000,#53,#54,#55,#56,#57)

4.9. <u>OleDoVerb</u> - Perform an object verb

This function to select a verb for an object. A verb is the action that may be performed on the object. For example, a verb may be Edit, Open, Play, etc., dependent on the object type. Normally, all objects have the verbs Edit and Open, where some objects have additional verbs like Play (Sound/Video).

If the object verb id is known it may be parsed directly in *par3* with *par2* set to 1. If not, the example from OleFillVerbMenu() may be used (the user may select the action to perform on the object from a menu).

The parameters *par5-par9* is required and may be obtained by the standard TRIO sub function GETINFO(). The parameters must be defined as a field format "8." (16/32 bit compatible). If wrongly defined the application may result in errors or unexpected results.

See OleFillVerbMenu

4.10. <u>OleSave</u> - Save object as embedded into file

This functions can save an object into a file. The object to save may be linked or embedded. If it is a link, only the file name is saved, as for an embedded object the entire object is saved. The saved object may be loaded again using the OleLoad() function.

OleLoad(#50,#51) /* Load the object

4.11. <u>OleLoad</u> - Load an embedded object from file

This function can load an object saved prior with OleSave().

See OleSave

4.12. <u>OleGetInfo</u> - Return information about object

The function returns information about the object. The information returned depends on the mode (*par2*), which can be one of the following: With mode 0 *par3* is not used. With mode 1 it will copy the file name of the object into *par3*

end

4.13. <u>OleSetInfo</u> - Set object Information

The function will set information about the object. The information to set depends on par2, which can be one of the following:

With mode 0 the object flags must be parsed in *par3*. The value may be one or more of the following values added together:

0 - No flags. The object is scaled according to the field box dimensions. 1 - The object uses its real dimensions. If the field box size is smaller than the object it is clipped. 2 - The vertical dimension is scaled according to the horizontal dimension of the field box. 4 - The horizontal dimension is scaled according to the vertical dimension of the field box.

Any object created uses flag 2 as default.

OleSetInfo(#50,0,0) /* Scale the object according to the field box

4.14. <u>OleMenuCreate</u> - Create a menu

The function may be used to create a Windows menu. When a menu has been created items may be added using the OleMenuAdd function.

When the menu is no longer needed it must be destroyed using the OleMenuDestroy function. If it is not, memory is not released!

if #52=3001 OleLinkToFile(#50,"c:/swtools/document.doc")

4.15. <u>OleMenuDestroy</u> - Destroy a menu

This function will free all memory used by a menu created with the OleMenuCreate() function. If any sub menu has been added to the menu parsed in *par1* it will be destroyed also.

See OleMenuAdd

4.16. <u>OleMenuAdd</u> - Add item to a menu

This function may be used to add items to a menu created with OleMenuCreate(). The *par2* controls the type of item to be added to the menu:

If *par2* is 0 a text item is added to the menu.

If *par2* is -1 a separator (dividing line) is added to the menu. With this mode *par3* and *par4* is not used.

If *par2* is greater than 0 it must be another unique menu id returned from the OleMenuCreate() function.

The parameter *par3* must be a unique menu item id, which will be returned when the user selects an item from the menu, and *par4* simply contains the text of the item.

OleMenuDestroy(#51) /* This will free all menus created in #51-53

4.17. <u>OleMenuSelect</u> - Select from menu at the current cursor location

The function activates a floating menu at the current cursor location on screen. The menu will remain active on screen until the user selects one of the items or clicks outside the menu.

See OleMenuAdd

4.18. <u>OleDialogCreate</u> - Standard dialog for creating embedded and linked objects

This function may be used to simplify the creation of an object. The *par2* is the language id, which controls the dialog text used.

if #51>1 OleEmbedded(#50,1,"",#51-1,0) /* Create embedded using the returned menu item id

4.19. <u>OleDialogFile</u> - Standard dialog for selecting file name

This function will let the user select a file name using the standard Windows file dialog. The window handle in *par1* may be obtained using the subfunction GETINFO().

The dialog requires two parameters to control the file name filters. The parameter *par2* is used for the extension of a file name, e.g. "ole" as extension will only list all files named "xxxxxxx.ole". The parameter *par3* is the definition of the filter including a filter description. For example, a filter may be defined as

"All files,*.*"

which defines a filter for the user to select with the description "All files" and the extension "*.*". It is important that the description and extension is separated by a comma, otherwise the dialog may fail to be displayed!

If the mode (*par4*) is 1 the file dialog will let the user enter a new file name or select an existing file name for save. If an existing file name is selected the user must answer yes in order to overwrite the file.

If a file name is parsed in *par5* and no value parsed in *par6* ("" empty text) the dialog will use the path from the file name in *par5*. For example, if *par5* is "c:/swtools/sheet.xls" the initial path will be "c:/swtools".

See OleGetInfo()

5. Technical specifications

5.1. Requirements

The OLE interface requires SW-Tools TRIO version 007.001 or higher. It will be supported in 16 and 32-bit versions.

5.2. Files installed

10.engSubfunction descriptions and interface for SW-Tools TRIO calculationsole-eng.hlpWindow on-line manualswo999xx.dllOLE Dynamic Link Library, where 999 is the major version number and xxis 16/32 bit

Figure list

1. Defining the supplier letter	6
2. Adding fields to print on letter	7
3. Defining the OLE 2.0 Object field	8
4. How to select the requested OLE object type	9
5. Selecting Microsoft Word as object type	. 10
6. Entering the content of the object in Microsoft Word	. 10
7. Layout of letter including the OLE object content	. 10
8. The printout of the supplier letter	. 11
9. Linking to an existing file	. 12
10. Simple supplier query in IQ	. 14
11. Free field defined as OLE object	. 15
12. Buttons to control the OLE object	. 16
13. Calculations for creating an OLE object	. 17
14. Calculations for saving the created OLE object	. 18
15. Calculations for performing actions on the OLE object	. 19
16. User creating an object for supplier no 205	. 20
17. User entering the note in Microsoft Word	. 21
18. IQ Query with user written note in Microsoft Word	. 21
19. On-line documentation of the functions	. 23

Index

A
Action16;20
C
Create 16;20;28;36;40
D
DATAMASTER13
E
Embedding1;3
G
GETINFO
I
IQ 3;4;13;14;20;21;45
L
Letter
Linking1;3;12;45
0
OLE
3;4;5;8;9;10;12;13;15;16;17;18;19;22;
23;24;25;26;27;43;44;45
OleAllocate25;26;27

OleDialogCreate	
OleDialogFile	
OleDoVerb	
OleEmbedded	
OleFillObjectMenu	
OleFillVerbMenu	
OleFree	
OleGetInfo	
OleLinkToFile	
OleLoad	
OleMenuAdd	. 36:37:38:39
OleMenuCreate	. 29:36:37:38
OleMenuDestrov	
OleMenuSelect	
OleSave	
0	
Z Query	21.45
D	
	2.4.5.10
	3;4;5;10
I	
TRIO3;6;13;21;22;27	7;28;31;43;44